

UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE QUITO – CAMPUS SUR

CARRERA DE INGENIERÍA DE SISTEMAS

**“ANÁLISIS, DISEÑO, IMPLEMENTACIÓN E IMPLANTACIÓN DE
UN SISTEMA DE INTEGRACIÓN DE DATOS PARA EL RASTREO DE
CARGA DE FLORES PARA EL SISTEMA TRACKING-CARGO DE LA
EMPRESA ENTICO CÍA. LTDA.”**

TESIS PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SISTEMAS

JOSÉ ALFREDO SÁNCHEZ GALLARDO

DIRECTORA ING. PATSY MALENA PRIETO VÉLEZ

QUITO, JUNIO 2013

DECLARACIÓN

Yo, José Alfredo Sánchez Gallardo, declaro bajo juramento que el trabajo aquí descrito es de mí autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en éste documento.

A través de la presente declaración cedo mis Derechos de Propiedad Intelectual correspondientes a éste trabajo, a la Universidad Politécnica Salesiana, según lo establecido en la Ley de Propiedad Intelectual, por su reglamento y por la normativa institucional vigente.

José Alfredo Sánchez Gallardo

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por José Alfredo Sánchez Gallardo bajo mi dirección.

Ing. Patsy Malena Prieto Vélez

Director de tesis

AGRADECIMIENTOS

Agradezco a Dios por cuidarme y bendecirme siempre al permitir que vea cada día el fruto de mi esfuerzo cotidiano.

Por medio de éste también me permito manifestar mis más sinceros agradecimientos hacia la Ing. Verónica Soria Maldonado, quien me brindó la oportunidad de adquirir experiencia en el área de sistemas y de ésta manera enfocarme directamente en un tema de actualidad e interés social como es el que desarrollo en éste proyecto, por su apoyo desinteresado y ayuda brindada durante éste tiempo.

A la Ing. Patsy Malena Prieto Vélez por ser mi guía y respaldarme durante todo éste tiempo.

DEDICATORIA

Dedico éste proyecto a mis padres Alfredo Sánchez y Nancy Gallardo por haberme enseñado que las metas se pueden alcanzar con esfuerzo y dedicación, la inexistencia de obstáculos insuperables, a ellos que me han acompañado en cada etapa de la vida guiándome por el camino adecuado, luchando a mi lado, brindándome siempre su apoyo, amor y confianza.

RESUMEN

La siguiente documentación tiene como fin demostrar cual es el funcionamiento de una agencia de carga de flores en el Ecuador y su relación con los documentos necesarios que necesitan los clientes y las fincas para poder exportar dicha carga.

Para ello, se ha implementado un software que le permite a la agencia de carga interactuar con ésta documentación ya que es necesaria para que la carga pueda salir. Dicho sistema está construido en plataforma JAVA evitando así la compra de licencias de determinado software de desarrollo.

Se han dividido capítulos en los que se conceptualiza a breves rasgos el proceso de exportación de flores; la funcionalidad que tiene la agencia dentro del país, la creación de pedidos entre la agencia – finca – cliente, la recepción de carga, la generación de la documentación necesaria para la salida de carga y desaduanización de la misma en el destino y la visualización en línea de los embarques solicitados por los clientes, teniendo la información en línea las 24 horas los siete días de la semana.

El sistema ha sido construido en base a la experiencia del usuario de agencia, con el cual, después de establecer un proceso, se procede con la construcción de dicho proceso en el programa.

Al final del proyecto, se han generado manuales de usuario sobre los cuales se puede entender el lineamiento de la página web, de la misma manera se han generado manuales técnicos que permitirán a los administradores del sitio, gestionar la funcionalidad del mismo.

ABSTRACT

The following documentation shows the operation of a flower freight in Ecuador and its relation to the necessary documents for clients and farms to export the load.

To do this, we have implemented a software that allows the agency to interact with documents necessary so the load can travel. This system is built on a JAVA platform, avoiding the purchase of licenses for certain software development.

The documentation has been split into chapters which are conceptualized to short view the flower export process, the functionality of the agency within the country, creating orders from the agency - farm - customer, receiving load generation documentation required for the departure of cargo and customs clearance and watch online of shipments requested by customers, having the information online 24 hours seven days a week.

The system has been built based on a user experience agency, after establishing a process, we proceed with the construction of the process in the program.

At the end of the project we have generated user manuals on which you can understand the guideline of the website, as well as technical manuals that will allow site administrators manage your functionality.

INIDICE GENERAL

1	CAPÍTULO 1. MARCO TEÓRICO	1
1.1	TEMA	1
1.2	PLANTEAMIENTO DEL PROBLEMA	1
1.2.1	DEFINICIÓN DE PROCESOS	2
1.2.1.1	PARA CADA CLIENTE, SE PROCEDE	3
1.2.1.2	PARA CADA FLORÍCOLA, SE PROCEDE	3
1.2.1.3	PARA CADA DESADUANIZADOR, SE PROCEDE	3
1.3	OBJETIVOS	4
1.3.1	OBJETIVO GENERAL	4
1.3.2	OBJETIVOS ESPECÍFICOS	4
1.4	JUSTIFICACIÓN DEL PROYECTO	6
1.5	ALCANCE DEL PROYECTO	7
1.5.1	MÓDULO DE TRANSFERENCIA DE DATOS	7
1.5.2	MÓDULO DE RECEPCIÓN DE DATOS	9
1.5.3	MÓDULO DE WEB	10
1.5.4	PERFIL DE USUARIO	12
1.5.5	PLATAFORMA DE DESARROLLO	12
1.6	METODOLOGÍA	14
1.6.1	METODOLOGÍA DE DISEÑO	14
1.6.1.1	LENGUAJE UNIFICADO DE MODELADO	14
1.6.1.2	DIAGRAMAS	15

INIDICE GENERAL

1.6.1.2.1	LOS DIAGRAMAS DE ESTRUCTURA	16
1.6.1.2.2	LOS DIAGRAMAS DE COMPORTAMIENTO	20
1.6.1.2.3	LOS DIAGRAMAS DE INTERACCIÓN	20
1.6.1.3	PROCESO UNIFICADO RACIONAL	21
1.6.1.3.1	PRINCIPIOS DE DESARROLLO	21
1.6.1.3.2	CICLO DE VIDA	23
1.6.1.4	PROJECT MANAGEMENT PROFESSIONAL, PMP	24
1.7	PROCESOS DE NEGOCIO	25
1.8	CLIENTE	25
1.8.1	BRÓKER	26
1.8.2	FLORÍCOLA	26
2	CAPÍTULO II. ANÁLISIS	27
2.1	PMP (PROFESIONALISMO EN LA ADMINISTRACIÓN DE PROYECTOS PROJECT MANAGEMENT PROFESSIONAL)	27
2.1.1	PMBOK	28
2.1.1.1	ESTRUCTURA DEL PMBOK	30
2.2	DIAGRAMAS BPMN (NOTACIÓN PARA EL MODELADO DE PROCESOS DE NEGOCIO)	35
2.2.1	ESPECIFICACIÓN DE REQUERIMIENTOS FUNCIONALES	36
2.2.1.1	MÓDULO PARA CLIENTES	36
2.2.1.2	MÓDULO DE BRÓKER	42
2.2.1.3	MÓDULO DE FLORÍCOLAS	42

INIDICE GENERAL

2.3	DIAGRAMAS UML	47
2.3.1	DIAGRAMA DE CASOS DE USO	47
2.3.2	DIAGRAMA DE ACTIVIDADES	50
3	CAPÍTULO III. DISEÑO	51
3.1	DIAGRAMAS UML	51
3.1.1	DIAGRAMA DE CLASES	51
3.1.2	DIAGRAMA DE OBJETOS	66
3.1.3	DIAGRAMA DE PAQUETES	67
3.1.4	DIAGRAMA DE COMPONENTES	68
3.1.5	DIAGRAMA DE SECUENCIA	69
3.2	DISEÑO DE BASES DE DATOS	75
3.2.1	MODELO FÍSICO	75
3.2.2	DICCIONARIO DE LA BASE DE DATOS	81
3.3	DISEÑO DE INTERFAZ GRÁFICA	87
3.3.1	FRAMEWORK CSS	87
3.3.2	FRAMEWORK CSS DEL SISTEMA	95
3.3.3	COMPONENTES RICHFACES	96
3.3.4	TEMPLATES FACELETS	98
4	CAPÍTULO IV. SEGURIDADES	101
4.1	ANÁLISIS DE AUTENTICACIÓN Y AUTORIZACIÓN	101
4.2	APLICACIÓN DE JAAS	102

INIDICE GENERAL

4.3	MANEJO DE FILTROS	104
4.3.1	MVC ("MODEL-VIEW-CONTROLLER")	104
4.3.2	CON JSP'S Y SERVLETS	104
4.3.3	JAVA BEANS	106
4.3.4	FILTROS	108
4.3.4.1	DEFINICIÓN DE UN FILTRO EN WEB.XML	109
4.3.4.2	CONFIGURACIÓN DE ARCHIVO ACCESACTIONFILTER	112
5	CAPÍTULO V. IMPLEMENTACIÓN Y PRUEBAS	114
5.1	PRUEBAS DE UNIDAD	114
5.2	PRUEBAS DE FUNCIONALIDAD	115
5.3	PRUEBAS DE ESTRÉS	117
5.3.1	HARDWARE	118
5.3.2	SOFTWARE	118
5.3.3	CASO DE PRUEBA	118
5.3.3.1	USUARIO PRUEBA	119
5.3.3.2	ESTADO DEL SERVIDOR	119
5.3.3.3	PROCEDIMIENTO A EVALUARSE	120
5.3.3.4	PETICIÓN A LA PÁGINA CON 100 HILOS	121
5.3.3.5	CONSUMO DE PROCESADOR CUANDO SE EJECUTA CONSULTA SOBRE VISUALIZACIÓN DE EMBARQUES	122
5.4	MANUALES TÉCNICOS Y PARA EL USUARIO	125

INIDICE GENERAL

5.4.1	MANUAL TÉCNICO	125
5.4.1.1	INSTALACIÓN DE POSTGRES 9.1	126
5.4.1.2	INSTALACIÓN JAVA	129
5.4.2	MANUAL DE USUARIO	131
6	CAPÍTULO VI. PLAN DE CONTINGENCIA	157
6.1	DEFINICIÓN DEL PLAN DE CONTINGENCIA LOCAL	157
6.2	DEFINICIÓN DEL PLAN DE CONTINGENCIA SERVIDOR DE APLICACIONES REMOTO	158
7	CAPÍTULO VII. CONCLUSIONES Y RECOMENDACIONES	160
7.1	CONCLUSIONES	160
7.2	RECOMENDACIONES	162
7.3	PROYECCION A FUTURO	163
	GLOSARIO DE TÉRMINOS	166
	BIBLIOGRAFIA	171

INDICE DE GRÁFICOS

Gráfico 1: Jerarquización de Diagramas UML; Autor: Sánchez Gallardo José.	15
Gráfico 2: Diagrama de Paquetes; Autor: Gutiérrez Damián.	18
Gráfico 3. Diagrama de Actividades; Autor: SparxSystem.	19
Gráfico 4. Componentes de un diagrama de casos de uso; Autor: Universidad de Chile.	20
Gráfico 5. Esfuerzo en actividades según fase del proyecto; Autor: Wikipedia. Org.	24
Gráfico 6. Proceso PMBOK; Autor: Sánchez Gallardo José.	28
Gráfico 7. Modelo Equivalente PMBOK; Autor: Sánchez Gallardo José.	29
Gráfico 8. Áreas del Conocimiento; Autor: Sánchez Gallardo José.	29
Gráfico 9. Áreas de Conocimiento; Autor: Sánchez Gallardo José.	32
Gráfico 10. Representación de Grupos de Procesos en PMBoK; Autor: Sánchez Gallardo José.	33
Gráfico 11. Representación de Grupos de Procesos en PMBoK Internos; Autor: Sánchez Gallardo José.	33
Gráfico 12. Procesos de Planeación del Proyecto según el PMBoK; Autor: Sánchez Gallardo José.	34
Gráfico 13. Diagrama BPMN Transmisión de Documentos; Autor: Sánchez Gallardo José.	35
Gráfico 14. Diagrama: Reservación; Autor: Sánchez Gallardo José.	36
Gráfico 15. Diagrama: Ingreso Bodega; Autor: Sánchez Gallardo José.	37
Gráfico 16. Diagrama: Etiquetas; Autor: Sánchez Gallardo José.	38
Gráfico 17. Diagrama: Documentación; Autor: Sánchez Gallardo José.	39
Gráfico 18. Diagrama: Corte de Guías; Autor: Sánchez Gallardo José.	41
Gráfico 19. Diagrama: Control de Calidad; Autor: Sánchez Gallardo José.	44
Gráfico 20. Diagrama: Coordinación; Autor: Sánchez Gallardo José.	46

Gráfico 21. Diagrama en Caso de Uso – Bróker; Autor: Sánchez Gallardo José.	47
Gráfico 22. Diagrama en Caso de Uso – Cliente; Autor: Sánchez Gallardo José.	48
Gráfico 23. Diagrama en Caso de Uso – Florícola; Autor: Sánchez Gallardo José.	49
Gráfico 24. Diagrama en Actividades; Autor: Sánchez Gallardo José.	50
Gráfico 25 a) Diagrama de Constantes; Sección: Constantes; Autor: Sánchez Gallardo José.	51
Gráfico 26. b) Diagrama de Constantes; Sección: Dao; Autor: Sánchez Gallardo José.	52
Gráfico 27.c) Diagrama de Constantes; Sección: Dao Ejb; Autor: Sánchez Gallardo José.	53
Gráfico 28. d) Diagrama de Constantes; Sección: Dao Jpa; Autor: Sánchez Gallardo José.	54
Gráfico. 29. e) Diagrama de Constantes; Sección: Excepciones; Autor: Sánchez Gallardo José.	55
Gráfico. 30. f) Diagrama de Constantes; Sección: Jdbc; Autor: Sánchez Gallardo José.	56
Gráfico. 31. g) Diagrama de Constantes; Sección: Modelo Consulta; Autor: Sánchez Gallardo José.	57
Gráfico. 32. h) Diagrama de Constantes; Sección: Modelo Dto; Autor: Sánchez Gallardo José.	58
Gráfico. 33. i) Diagrama de Constantes; Sección: Modelo Persistencia; Autor: Sánchez Gallardo José.	59
Gráfico: 34. j) Diagrama de Constantes; Sección: Servicios; Autor: Sánchez Gallardo José.	60
Gráfico: 35. m) Diagrama de Constantes; Sección: Servicios – Implementación; Autor: Sánchez Gallardo José.	61
Gráfico: 36. n) Diagrama de Constantes; Sección: Servicios – Interfaces; Autor: Sánchez Gallardo José.	62
Gráfico: 37. o) Diagrama de Constantes; Sección: Utilitarios; Autor: Sánchez Gallardo José.	63

Gráfico 38. p) Diagrama de Constantes; Sección: Web; Autor: Sánchez Gallardo José.	64
Gráfico 39. q) Diagrama de Constantes; Sección: Web Service; Autor: Sánchez Gallardo José.	65
Gráfico 40. Diagrama de Objetos; Sección: Refrendos; Autor: Sánchez Gallardo José.	66
Gráfico 41. Diagrama de Paquetes; Autor: Sánchez Gallardo José.	67
Gráfico 42. Diagrama Componentes; Autor: Sánchez Gallardo José.	68
Gráfico 43. a) Diagrama de Secuencia; Sección: Administración de Parámetros; Autor: Sánchez Gallardo José.	69
Gráfico 44. b) Diagrama de Secuencia; Sección: Autenticación; Autor: Sánchez Gallardo José.	70
Gráfico 45. c) Diagrama de Secuencia; Sección: Broker – Consulta; Autor: Sánchez Gallardo José.	71
Gráfico 46. d) Diagrama de Secuencia; Sección: Cliente – Consulta; Autor: Sánchez Gallardo José.	72
Gráfico 47. e) Diagrama de Secuencia; Sección: Consulta – Embarque; Autor: Sánchez Gallardo José.	73
Gráfico 48. Diagrama de Secuencia; Sección: Florícola – Consulta; Autor: Sánchez Gallardo José.	74
Gráfico 49. literal a) Tablas de Negocio; Autor: Sánchez Gllardo José.	77
Gráfico 50. literal b) Tablas de Parámetros; Autor: Sánchez Gallardo José.	78
Gráfico 51. literal c) Relación awb_reservacionmensual_tbl con parámetros; Autor: Sánchez Gallardo José.	79
Gráfico 52. literal d) Relación awb_cabeceraexportacion_tbl con parámetros; Autor: Sánchez Gallardo José.	80
Gráfico 53. literal e) Relación awb_detalleexportacion_tbl con parámetros; Autor: Sánchez Gallardo José.	80
Gráfico 54. Esquema de solicitud de procesamiento de flujo; Autor: Junta de Andalucía.	91
Gráfico 55. Elementos RichFaces; Autor: Junta de Andalucía.	92
Gráfico 56. Filtros en Aplicación Web; Autor: Sánchez Gallardo José.	94

Gráfico 57. Archivo de Configuración: login.xhtml; Autor: Sánchez Gallardo José.	103
Gráfico 58. Modificación al código de JSP's y/o Servlets; Autor: Osmia Latina.	105
Gráfico 59. Java Bean; Autor: Osmia Latina.	107
Gráfico 60. Definición de un filtro en web.xml; Autor: Osmia Latina.	109
Gráfico 61. Configuración del filtro en el archivo web.xml; Autor: Sánchez Gallardo José.	111
Gráfico 62. Configuración de archivo AccesactionFilter_Parte a; Autor: Sánchez Gallardo José.	112
Gráfico 63. Configuración de archivo AccesactionFilter_Parte b; Autor: Sánchez Gallardo José.	113
Gráfico 64. Pruebas de Unidad; Autor: Sánchez Gallardo José.	115
Gráfico 65. a) Pruebas de Funcionalidad; Autor: Sánchez Gallardo José.	116
Gráfico 66. b) Pruebas de Funcionalidad; Autor: Sánchez Gallardo José.	116
Gráfico 67. c) Pruebas de Funcionalidad; Autor: Sánchez Gallardo José.	117
Gráfico 68. Consulta de Prueba; Autor: Sánchez Gallardo José.	118
Gráfico 69. Estado del Servidor; Autor: Sánchez Gallardo José.	119
Gráfico 70. Procedimiento a Evaluarse; Autor: Sánchez Gallardo José.	120
Gráfico 71. Resultado de petición a la página con 100 hilos; Autor: Sánchez Gallardo José.	121
Gráfico 72. a) Consumo de procesador en visualización de embarques; Autor: Sánchez Gallardo José.	122
Gráfico 73. b) Resultado en el procesador sobre la visualización de embarques, 1; Autor: Sánchez Gallardo José.	123
Gráfico 74. c) Resultado en el procesador sobre la visualización de embarques, 2; Autor: Sánchez Gallardo José.	124
Gráfico 75. d) Resultado en el procesador sobre la visualización de embarques en long; Autor: Sánchez Gallardo José.	125

Gráfico 76. a) Secciones Base y Updates Modificadas; Autor: Sánchez Gallardo José.	126
Gráfico 77. b) Enlace de última versión de la rama 9.1; Autor: Sánchez Gallardo José.	127
Gráfico 78. c) Configuración del Archivo pg_hba.conf; Autor: Sánchez Gallardo José.	128
Gráfico 79. d) Entrar a la base de datos desde el cliente pg_admin instalado en la máquina; Autor: Sánchez Gallardo José.	129
Gráfico 80. a) Verificación de la Instalación en el repositorio rpmforge; yum repolist.; Autor: Centosni.	130
Gráfico 81. b) Ejecución de Sintaxis; Autor: Centosni.	130
Gráfico 82. a) Verificación de Instalación JAVA-version.; Autor: Centosni.	131
Gráfico 83. Utilización de comandos en caso de caída de Postgres; Autor: Sánchez Gallardo José.	131
Gráfico 84. a) Página Principal Agencia de Carga TransInternacional; Autor: TRANS Internacional Cargo.	132
Gráfico 85. b) Página de Consulta de Datos de los Usuarios; Autor: TRANS Internacional Cargo.	133
Gráfico 86. c) Acceso Específico de Clientes – link Customers; Autor: TRANS Internacional Cargo.	134
Gráfico 87. d) Visualización Acceso de Clientes; Autor: Sánchez Gallardo José.	134
Gráfico 88. e) Visualización de las 3 opciones de Consulta; Autor: Sánchez Gallardo José.	135
Gráfico 89. f) Previous Shipmet – Ultimo Embarque; Autor: Sánchez Gallardo José.	136
Gráfico 90. g) Previous Shipmet – Datos Principales del usuario; Autor: Sánchez Gallardo José.	136
Gráfico 91. h) Previous Shipmet – Visualización de Documentos del Cliente; Autor: Sánchez Gallardo José.	137
Gráfico 92. i) Previous Shipmet – Descarga de Documentos del Cliente I; Autor: Sánchez Gallardo José.	137

Gráfico 93. j) Previous Shipmet – Visualización de datos principales del Cliente; Autor: Sánchez Gallardo José.	138
Gráfico 94. k) Previous Shipmet – Opciones de Descarga de Documentos del Cliente; Autor: Sánchez Gallardo José.	139
Gráfico 95. l) Previous Shipmet –Descarga de Documentos del Cliente II; Autor: Sánchez Gallardo José.	139
Gráfico 96. m) Previous Shipmet –Descarga de Documentos del Cliente III; Autor: Sánchez Gallardo José.	140
Gráfico 96. n) Search by Date – Selecccion de Rango de fechas 1; Autor: Sánchez Gallardo José.	141
Gráfico 97. o) Search by Date – Selecccion de Rango de fechas 2; Autor: Sánchez Gallardo José.	141
Gráfico 98. p) Search by Date – Parámeto de fechas 3; Autor: Sánchez Gallardo José.	142
Gráfico 99. q) Search by AWB; Autor: Sánchez Gallardo José.	143
Gráfico 100. r) Search by AWB – Guía consultada de Acuerdo al número; Autor: Sánchez Gallardo José.	143
Gráfico 101. s) link CUSTOMS BROKERS; Autor: TRANS Internacional Cargo.	144
Gráfico 102. t) link CUSTOMS BROKERS – Visualización de Acceso; Autor: Sánchez Gallardo José.	145
Gráfico 103. u) Visualización de las 4 opciones de Consulta; Autor: Sánchez Gallardo José.	146
Gráfico 104. v) Consulta por Ultimo Embarque; Autor: Sánchez Gallardo José.	147
Gráfico 105. w) Consulta por Ultimo Embarque - Descarga de Documentos; Autor: Sánchez Gallardo José.	147
Gráfico 106. x) Consulta por Ultimo Embarque - Visualización de Documentos; Autor: Sánchez Gallardo José.	148
Gráfico 107. y) Consulta por Búsqueda de Awb por fecha de vuelo; Autor: Sánchez Gallardo José.	149
Gráfico 109. z) Consulta por Awb por Rango de Fechas; Autor: Sánchez Gallardo José.	150

Gráfico 110. a1) Consulta por Búsqueda por número de Awb específico; Autor: Sánchez Gallardo José.	151
Gráfico 111. b1) link EXPORTERS - Específico para las Florícolas; Autor: TRANS Internacional Cargo.	152
Gráfico 112. b1) Autenticación de Datos de las Florícolas; Autor: Sánchez Gallardo José.	153
Gráfico 113. c1) EXPORTERS - Visualización de las opciones de Consulta; Autor: Sánchez Gallardo José.	153
Gráfico 114. d1) EXPORTERS - Consulta por Búsqueda de Awb por fecha de Coordinación; Autor: Sánchez Gallardo José.	154
Gráfico 115. e1) EXPORTERS –Visualización del Detalle de Guía; Autor: Sánchez Gallardo José.	154
Gráfico 116. f1) EXPORTERS – Consulta por Awb por Rango de Fechas de Coordinación; Autor: Sánchez Gallardo José.	155
Gráfico 117. g1) EXPORTERS – Consulta por Búsqueda por número de Awb específico; Autor: Sánchez Gallardo José.	156

INDICE DE TRABLAS

Tabla 1. awb_reservacionmensual_tbl; Autor: Sánchez Gallardo José.	81
Tabla 2. awb_cabeceraexportacion_tbl; Autor: Sánchez Gallardo José.	82
Tabla 3. awb_detalleexportacion_tbl; Autor: Sánchez Gallardo José.	83
Tabla 4. awb_detalleingresobodega_tbl; Autor: Sánchez Gallardo José.	84
Tabla 5. awb_documentoexportacion_tbl; Autor: Sánchez Gallardo José.	84
Tabla 6. awb_rutaawb_tbl; Autor: Sánchez Gallardo José.	85
Tabla 7. pa_aerolinea_tbl; Autor: Sánchez Gallardo José.	85
Tabla 8. pa_broker_tbl; Autor: Sánchez Gallardo José.	86
Tabla 9. pa_cliente_tbl; Autor: Sánchez Gallardo José.	86
Tabla 10. pa_floricola_tbl; Autor: Sánchez Gallardo José.	86
Tabla 11. pa_tipocliente_tbl; Autor: Sánchez Gallardo José.	87
Tabla 12. pa_contador_tbl; Autor: Sánchez Gallardo José.	87
Tabla 13. Framework utilizado para el sistema; Autor: Sánchez Gallardo José.	95
Tabla 14. Componentes utilizados; Autor: Sánchez Gallardo José.	97
Tabla 15. Plantillas utilizadas para páginas JSF; Autor: Sánchez Gallardo José.	99
Tabla 16. Configuración filtro; Autor: Sánchez Gallardo José.	100

CAPÍTULO I

MARCO TEÓRICO.

1.1 Tema

“Análisis, Diseño, Implementación e Implantación de un sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.”

1.2 Problema

ENTICO Cía. Ltda., es una empresa de servicios tecnológicos y brinda servicios de desarrollo de software de manera eficiente, oportuna y de alta calidad, la cual se encuentra constituida bajo las leyes y reglamento de la república del Ecuador.

Actualmente se encuentra en desarrollo un sistema para el rastreo de carga perecible llamado TRACKING-CARGO, para la agencia de carga TRANSINTERNACIONAL CARGO.

1.2.1 Definición de Procesos

La agencia de carga registra la reservación del vuelo en la línea aérea dependiendo del destino final del embarque, durante éste proceso se registra los datos del cliente final, ruta de vuelo, y cupo de carga a ser enviados, éste proceso se lo realiza con semanas y meses de anticipación.

- **Problema:** El cliente final requiere ésta información constantemente para planificar sus ventas;

- Un cliente puede trabajar con varias florícolas, solo en la provincia de Pichincha existen alrededor de 600 florícolas, cuando llega el día del embarque cada una de estas florícolas coordina la carga que va a enviar durante el día, este proceso se lo realiza mediante una llamada telefónica;
- **Problema:** Durante éste proceso el cliente necesita monitorear constantemente las florícolas que van a embarcar la carga, cantidad, hora. Esta información es necesaria en caso de que alguna florícola no lo realice, con el detalle del embarque, el cliente puede tener un estimado de la carga que se debe receptar en los cuartos fríos en la noche, de la misma forma las líneas telefónicas se saturan cuando existe una gran cantidad de embarques, y las florícolas deben esperar mucho tiempo en el teléfono para realizar una coordinación de embarques;
- Cuando el vuelo ha sido confirmado y la florícola ha realizado la coordinación durante el día, se procede a receptar la carga a ser exportada en los cuartos fríos de la agencia de carga, este proceso empieza a partir de las 5pm. Se registra el detalle, temperatura, tipo de caja, documentos, datos del camión, conductor, etc. por cada florícola;
- **Problema:** El cliente necesita saber constantemente la cantidad de carga receptada, y sobre todo si existió alguna novedad ya sea por parte de la florícola o por la agencia de carga;
- Durante todo este proceso, el mayor problema es la comunicación y la diferencia de uso horario ya que todos los clientes se encuentran en el exterior, y necesitan interactuar constantemente con sus embarques, tomando en cuenta que son productos perecibles y el cliente debe tomar decisiones rápidas en caso de encontrar algún inconveniente; y,
- Esta interacción se lo realiza mediante email, teléfono, fax, y es muy costoso para la agencia de carga informar a todos sus clientes constantemente.

1.2.1.1 Para cada cliente, se procede:

- Digitalización de todas las guías que vuelan, y envío por mail, a cada comprador;
- Digitalización de todos los Certificados de Origen, y envío por mail, a cada comprador;
- Digitalización de todos los Fitos Sanitarios, y envío por mail, a cada comprador;
- Digitalización de cada una de las facturas de las florícolas, y envío por mail, a cada comprador; y,
- Confirmación telefónica o mediante fax con información de las rutas de vuelo de la aerolínea.

1.2.1.2 Para cada florícola, se procede:

- Digitalización de cada una de las guías Hawb¹ de las florícolas, y envío por mail, a cada comprador;
- Digitalización de todas las guías que vuelan, y envío por mail, a cada desaduanizador;
- Digitalización de todos los Certificados de Origen², y envío por mail, a cada desaduanizador;
- Digitalización de todos los Fitos Sanitarios³, y envío por mail, a cada desaduanizador;

¹ **HAWB: (HOUSE AIR WAYBILL)** conocimiento aéreo (de casa) o sea, el propio. Puede tener cualquier formato e información y sirve como guía y justificante de un envío en concreto entre un exportador y un importador específico. Por razones prácticas y a veces por imperativo legal, se utiliza el mismo formato y condiciones que vienen reflejadas en los conocimientos aéreos de las líneas aéreas (IATA) , sin embargo en éste documento no hace falta ni el código IATA, ni una tarifa “oficial” u otros requisitos mandatorios en el conocimiento aéreo IATA.

² **CERTIFICADOS DE ORIGEN:** El Certificado de Origen acredita frente a la autoridad aduanera de importación, el país de fabricación o producción de la mercancía, como también lo acreditan otros documentos de origen (EUR y FORM A).

³ **FITOS SANITARIOS:** Es el documento probatorio de la condición de salubridad en que se encuentran determinados productos vegetales exportables. En Ecuador los certificados fitosanitarios son otorgados por las dependencias del Ministerio de Agricultura y Ganadería.

- Digitalización de todas las guías que vuelan, y envío por mail, a cada de las florícolas; y,
- Digitalización de cada una de las guías Hawb de las florícolas, y envío por mail, a cada una.

1.2.1.3 Para cada desaduanizador, se procede:

- Digitalización de todas las guías que vuelan, y envío por mail, a cada comprador;
- Digitalización de todos los Certificados de Origen, y envío por mail, a cada comprador; y,
- Digitalización de todos los Fitos Sanitarios, y envío por mail, a cada comprador.

1.3 Objetivos

1.3.1 Objetivo General

Automatizar el proceso de rastreo de carga de flores para los clientes de una agencia de carga, desde el momento que se realiza la coordinación de carga, recepción, documentación, embarque y envío de la carga al destino final, para que los clientes tengan una información veraz y digitalizada sin la necesidad de esperar el horario de oficina en el Ecuador.

1.3.2 Objetivos Específicos

- Registrar la reservación de vuelo para el envío de la carga en la aerolínea;
- Digitalizar los documentos de exportación para que el cliente pueda desaduanizar la carga en el exterior;

- Construir un sistema con la información necesaria para que el cliente pueda rastrear el estado de su carga;
- Reducir costos operativos de la agencia de carga poniendo a disponibilidad la información de embarques en un sistema automatizado;
- Reducir costos de infraestructura tecnológica utilizando servidores en centros de acopio de datos;
- Asegurar la disponibilidad de los sistemas informáticos con sistemas redundantes para que el cliente pueda ver su información las 24 horas al día;
- Reducir costos de licenciamiento de software utilizando herramientas Open Source que permitan la comunicación entre sistemas de una forma segura y a bajos costos;
- Diseñar una metodología de transferencia de datos mediante Web Services que permita determinar cualquier error de transferencia de datos entre sistemas; y,
- Implementar una arquitectura J2EE para el manejo transferencia de datos y una simulación de J2EE mediante Spring y JPA⁴ para la recepción y consulta de datos de los clientes externos.

1.4 Justificación del Proyecto

Para cumplir con los objetivos mencionados anteriormente se desarrollarán tres módulos que interactúan de manera asíncrona y síncrona para que el cliente pueda contar con la información de sus embarques:

⁴ **JPA:** Java Persistence API, más conocida por sus siglas JPA, es la API de persistencia desarrollada para la plataforma Java EE. Es un framework del lenguaje de programación Java que maneja datos relacionales en aplicaciones usando la Plataforma Java en sus ediciones Standard (Java SE) y Enterprise (Java EE). El objetivo que persigue el diseño de esta API es no perder las ventajas de la orientación a objetos al interactuar con una base de datos (siguiendo el patrón de mapeo objeto-relacional).

Disponibilidad y acceso al sistema con estándares internacionales: Para una disponibilidad continua se desarrollará un módulo en internet esta aplicación estará en un servidor en Canadá que tenga una red de acceso T1⁵, para que los clientes tengan una mayor velocidad con estándares internacionales;

Seguridad de acceso a datos: Un módulo del sistema TRACKING-CARGO permitirá de forma asíncrona transferir información de la base de datos local a una base de datos del sistema Web mediante web services, de esta manera solo existe transferencia de datos, y el sistema Web es el encargado de presentar la información al cliente final, el servidor local nunca es visto de forma pública;

Reducción de costos: Al transferir información a una central de datos la agencia de carga reduce costos en infraestructura como servidores web y de aplicaciones, reduce costos de comunicación ya que no necesita un ancho de banda grande; y,

Arquitectura escalable: El sistema contará con una arquitectura que le permita escalar conforme la empresa crezca en número de transacciones, datos y accesos.

1.5 Alcance del Proyecto

Las agencias de carga trabajan con el sistema TRACKING CARGO en el cual registran la reservación de la aerolínea de vuelo de la carga, el día de exportación registran el detalle de la carga de cada florícola para cada cliente, en la noche se realiza la recepción de la carga en bodega, y se empieza a realizar todos los trámites y documentos para realizar la exportación de la carga, hasta el siguiente día que se embarca la carga en las diferentes aerolíneas.

⁵ **RED DE ACCESO T1:** La serie T de Estado Unidos, Canadá y Japón, 24 canales DSO forman una conexión llamada DS1 o T1 a una velocidad de 1.544 Mbps. Una trama T1 se compone de 193 bits, de los cuales 192 son para datos y 1 bit extra llamado bit de trama es utilizado para sincronizar. las series T y E operan en la capa 1 y 2 del modelo OSI, éstas utilizan TDM para asignar franjas de tiempo a la transmisión de datos, el medio de transmisión más común para redes T y E es el cable de cobre de par trenzado y fibra óptica y el costo de éste tipo de redes es moderado.

Para que el cliente pueda rastrear todo lo que sucede con su embarque se desarrollarán tres módulos que permitan transferir la información a un sistema en un servidor en una central de datos fuera del país, una aplicación que recepte la transmisión de datos, y una aplicación web que le permita al cliente poder rastrear los diferentes estados de la carga.

1.5.1 Módulo de transferencia de datos

Este módulo se encarga de armar las tramas de datos para la transferencia de datos del sistema TRACKING-CARGO al sistema consolidado de consulta y rastreo de carga en una central de datos fuera del país.

- a. **Fecha de exportación:** Determina la fecha de exportación, para la apertura de la transferencia de información;
- b. **Administración Aerolíneas:** Determina los datos de la aerolínea en donde se embarca la carga;
- c. **Administración de Clientes:** Determina los datos de los clientes y claves de acceso a la aplicación web para el rastreo de la carga;
- d. **Administración de florícolas:** Determina los datos de los exportadores de carga;
- e. **Administración de brókers⁶:** Administra los datos de los brókers o agentes afianzados que se encargan de la desaduanización de la carga en los aeropuertos de destino;
- f. **Reservación de línea de vuelo:** Administra todos los datos de reservación de cupo de aerolínea, que son:
 - Número de guía aérea;
 - Fecha de vuelo;

⁶ **BRÓKERS:** El bróker es aquel individuo que se desempeña como intermediario en las operaciones de tipo financiero. También conocido como corredor y agente, actuará como nexo entre el comprador y el vendedor en cuestión y por ese trabajo cobrará una comisión determinada.

- Cupo;
- Aeropuerto de origen;
- Aeropuerto de destino;
- Tipo de pago; y,
- Número de vuelo.

g. Detalle de Coordinación: Administra los datos del detalle de la carga coordinada previamente por cada uno de los exportadores:

- Ruta de vuelo;
- Paletizadora⁷;
- Detalle de exportadores;
- Detalle de carga por cada exportador; y,
- Número de referendo de cada exportador.

h. Detalle de recepción de carga: Administra los datos de toda la carga receptada por la agencia de carga y que será embarcada en las diferentes aerolíneas:

- Número de ingreso a bodega;
- Camión de embarque;
- Conductor;
- Fecha y hora de recepción de carga; y,
- Cantidad y detalle de carga.

i. Documentos de desaduanización: Digitalización de todos los documentos de exportación realizados por la agencia de carga para cada guía de exportación.

⁷ **PALETIZADORA:** Se trata de una máquina que combina componentes mecánicos y eléctricos con la finalidad de colocar productos generalmente almacenados en cajas, sacos, tambores, entre otros, sobre un palé, que puede ser de madera, metal o plástico para la conformación de una estiba.

1.5.2 Módulo de recepción de datos

Se encarga de recibir la información transmitida por el módulo de transferencia, al receptor los datos automáticamente se encarga de crear, actualizar y responder cualquier error en la transferencia de los mismo, para lo cual está dividido en tres sub-módulos que son:

- a. Configuración de parámetros:** Se encarga de registrar todos los parámetros corporativos del módulo de transferencia, si encuentra los datos los actualiza caso contrario los crea:
 - Clientes;
 - Florícolas;
 - Bróker;
 - Aerolínea;
 - País; y,
 - Ciudad.

- b. Detalle de exportación:** Registra o actualiza los datos de todo el detalle de la exportación a ser visualizada por los actores de la aplicación web:
 - Detalle de la reservación de vuelo;
 - Detalle de cada exportador;
 - Detalle de cada ingreso a bodega por exportador; y,
 - Detalle de la ruta y aerolínea de vuelo.

- c. Documentos de exportación:** Administra toda la documentación digitalizada para la desaduanización de la carga en el aeropuerto de destino final de la misma.

1.5.3 Módulo de Web

Este módulo permite a los clientes, florícolas, brókers rastrear el estado de la carga con toda la información que el módulo de transferencia y recepción realizan, este módulo se divide en:

a. Módulo web clientes: A este módulo solamente ingresan los clientes que tengan un usuario de ingreso y contraseña de ingreso:

- *Registro de Reservación carga:* Despliega un listado completo de las reservaciones que tiene el cliente, la consulta es por rango de fechas.
- Rastreo de Tracking de embarques que tiene el cliente, la consulta es por:
 - Último embarque;
 - Embarques por fecha;
 - Número de guía aérea;
 - Despliega documentos de exportación anexados al embarque, las opciones son:
 - Último embarque;
 - Por número de guía aérea; y,
 - Por fecha de embarque;
- Monitoreo de carga permite llevar un rastreo del vuelo post salida, por cada uno de los destinos, las opciones son:
- Último embarque; y,
- Número de guía aérea.

b. Módulo web bróker: Este módulo es para los brókers, la información que presenta es solamente los datos de la guía aérea, no el detalle de exportadores:

- Rastreo de embarques donde se detalla la cabecera de la guía aérea por: último embarque, fecha de embarque, número de guía aérea;

- Detalla los documentos de exportación que el cliente permite que el bróker tenga acceso; y,
 - Monitoreo de carga en donde se lleva un rastreo del vuelo por cada destino confirmando la llegada o demora al mismo por último embarque, fecha de embarque, número de guía aérea.
- c. **Módulo web florícola:** Este módulo tiene acceso las florícolas que se les autoriza y tienen clave de acceso al sistema:

- *La florícola podrá realizar coordinación* es decir, reservar una cantidad específica de cajas a ser enviadas al cliente; la que entra a una cola de coordinaciones para que en la agencia se acepte o no la coordinación y mediante email el sistema confirma la coordinación a la finca. Las florícolas no podrán ver todos los clientes por razones de seguridad. La agencia de carga mediante una opción podrá autorizar a la florícola a coordinar, así podrá abrir el sistema a todas las florícolas o a unas florícolas específicas;
- *Visualización del Detalle de Embarque*, se despliega los datos de la guía aérea, pero solamente la coordinación de esa florícola no de las demás, las opciones son: último embarque, fecha de embarque, número de guía aérea;
- *Registro de Guías certificadas generadas*; tanto los recibos semanales, como las facturas mensuales, así la florícola podrá tener un estado de cuenta completo de las guías certificadas que la agencia entrega;
- Registrar FITO sanitario vigente, y anexar una imagen digitalizada del mismo; y,
- Monitoreo del embarque a donde se traslada la carga por: último embarque, fecha de embarque, número de guía aérea.

1.5.4 Perfil de usuario

1.5.4.1 Administrador: Puede revisar toda información de los clientes, brókers, florícolas;

1.5.4.2 Florícola: Puede revisar la información solamente de las guías aéreas en las que se encuentre como exportador, realizar coordinación de carga, y detalle de ingresos a bodega por cada coordinación realizada;

1.5.4.3 Cliente: Puede revisar el detalle de todas las guías aéreas asignadas, por fecha de exportación, esta información incluye, detalle de coordinación, detalle de ingresos a bodega, ruta de vuelo, y toda la documentación para la desaduanización de la carga en el destino final; y,

1.5.4.4 Bróker: Puede revisar el resumen de todas las guías aéreas asignadas, por fecha de exportación, esta información incluye, ruta de vuelo, y toda la documentación para la desaduanización de la carga en el destino final.

1.5.5 Plataforma de desarrollo

Todos los módulos serán desarrollados en JAVA 5.0 con tres diferentes arquitecturas:

1.5.5.1 Módulo de transferencia de datos: desarrollado en una arquitectura J2EE:

- Session Beans (Staleless);
- Entity Bean (JPA);
- Quarts⁸ para procesos asíncronos;
- Base de datos Postgres 8.3 –versión 2012-;
- JSF 1.2 (capa Web) –versión 2012-;
- Componentes Rich Faces 3.3.2 (Interface gráfica);
- JAXB (Web Services) –versión 2012-; y,
- Servidor de aplicaciones JBOSS 5.0.1.

⁸ **QUARTS:** es una unidad de volumen igual a un cuarto de galón, pintas, dos o cuatro tazas.

1.5.5.2 Módulo de recepción de datos: Arquitectura J2EE con una implementación de Spring⁹ y JPA:

- Capa de servicios (Spring);
- Capa de datos (JPA);
- Axis (Web Services);
- Base de datos Postgres 8.3; y,
- Servidor de aplicaciones Tomcat 5.5.

1.5.5.3 Módulo Web: Arquitectura J2EE con una implementación de spring y JPA:

- Capa de servicios (Spring);
- Capa de datos (JPA);
- JSF 1.2 Capa Web;
- Componentes Rich Faces 3.3.2 (interfaz gráfica);
- Base de datos Postgres 8.3; y,
- Servidor de aplicaciones Tomcat 5.5.

1.5.5.4 Estándar de servicios Web: Para el uso de Web services se implementa una arquitectura con XML, y Dom4j que permita la administración de datos mediante metadatos de XML.

1.5.5.5 Estándar de desarrollo: Para la codificación de Java se utilizará Code Conventions de Java.

⁹ **SPRING:** Spring es un framework java, ligero, cuyo concepto principal es implementar el patrón Inversión del Control, también conocido como Inyección de Dependencias. Spring, como veremos nos permite disfrutar de servicios de middleware que de otra forma estaríamos obligados a usar mediante contenedores EJB. Spring nos permite diseñar y escribir código en capas, abstrayéndonos de conceptos cruzados como la búsqueda de componentes, el manejo de transacciones, la seguridad, el uso directo de sentencias SQL, el manejo de excepciones, y otros.

1.5.5.6 Conectividad: La conectividad que debe existir entre los diversos módulos del programa se realizará mediante Web services.

1.6 Metodología

1.6.1 Metodología de diseño

1.6.1.1 Lenguaje Unificado De Modelado

Lenguaje Unificado de Modelado –UML- es el lenguaje de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG¹⁰ -Object Management Group-, que es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

UML es un "lenguaje de modelado" para especificar o para describir métodos y procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema, para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software entregando gran variedad de formas para dar soporte a una metodología de desarrollo de software -tal como el Proceso Unificado Racional o RUP-¹¹, pero no especifica en sí mismo qué metodología o proceso usar.

¹⁰ **OMG:** que es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

¹¹ **RUP:** El RUP (Proceso Unificado de Rational), es una metodología que busca mejorar las practicas que se implementan en el desarrollo de software, basándose en requerimientos comprobados a nivel comercial, y que, en el ámbito de oferta y demanda actual cumpla con los requerimientos obtenidos.

1.6.1.2 Diagramas:

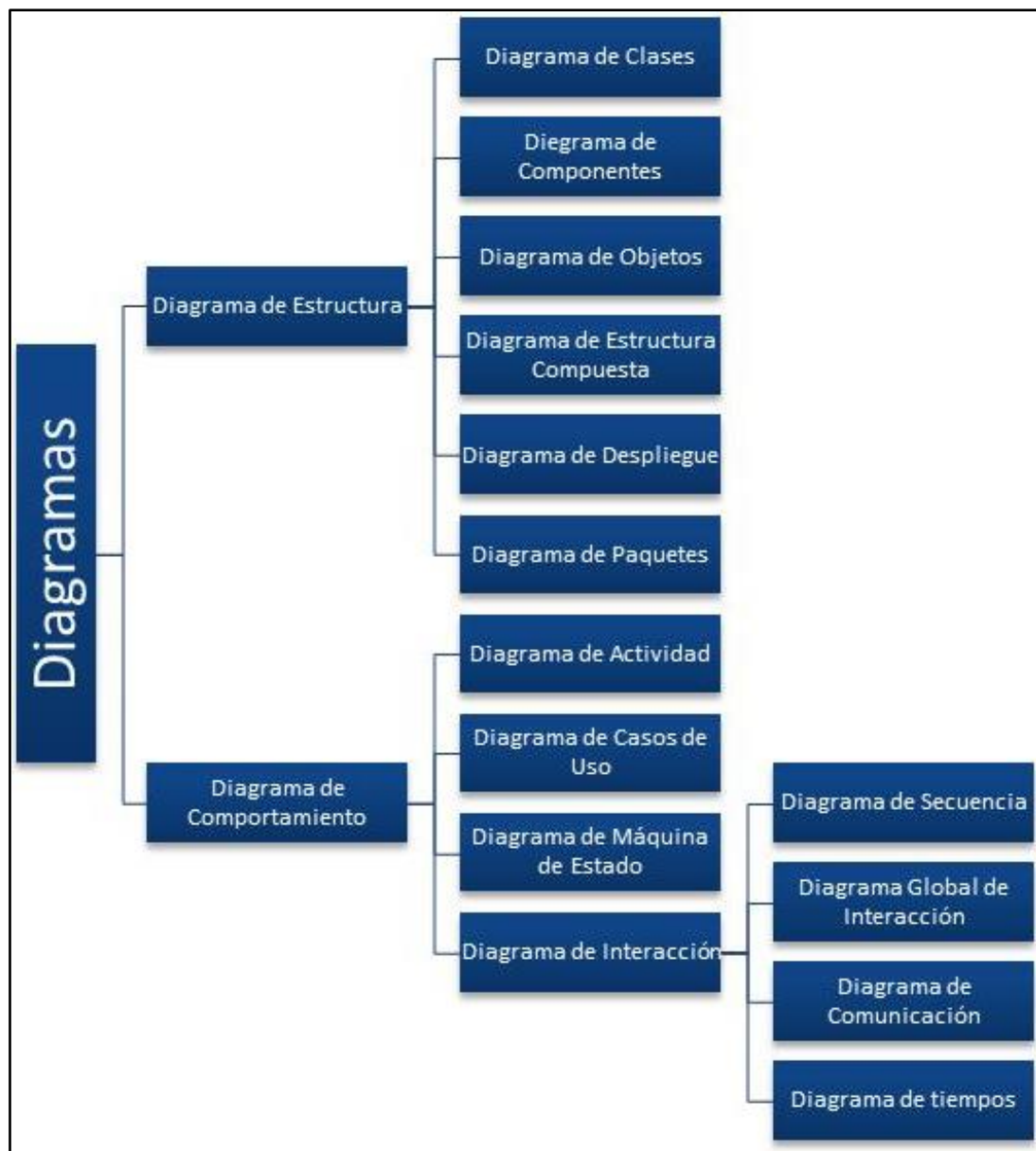


Gráfico 1: Jerarquización de Diagramas UML.

Autor: Sánchez Gallardo José.

En UML 2.0 hay 13 tipos diferentes de diagramas. Para comprenderlos de manera concreta, a veces es útil categorizarlos jerárquicamente.

1.6.1.2 .1. Los Diagramas de Estructura: enfatizan los elementos que deben existir en el sistema modelado, es decir éstos nos muestran aquellas partes o componentes de una especificación que sean independientes del tiempo:

- a. Diagrama de clases;
- b. Diagrama de componentes;
- c. Diagrama de objetos;
- d. Diagrama de estructura compuesta (UML 2.0);
- e. Diagrama de despliegue; y,
- f. Diagrama de paquetes.

a. Diagrama de clases:

Un diagrama de clases es de tipo estático cuya característica principal es describir la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Son utilizados primordialmente durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargarán del funcionamiento y la relación entre uno y otro, sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de contenimiento.

Definiciones:

- **“Propiedades:** Una propiedad es un miembro que ofrece un mecanismo flexible para leer, escribir o calcular el valor de un campo privado. Las propiedades pueden utilizarse como si fuesen miembros de datos públicos, aunque en realidad son métodos especiales denominados descriptores de acceso. De este modo, se puede obtener acceso a los datos con facilidad, a la vez que se promueve la seguridad y flexibilidad de los métodos.” (Library, 2013);
- **Operaciones;** comúnmente llamados métodos, son aquellas actividades o verbos que se pueden realizar con o para éste objeto, como por ejemplo

abrir, cerrar, buscar, cancelar, acreditar, cargar. El nombre de una operación se escribe con minúsculas si consta de una sola palabra. Si el nombre contiene más de una palabra, cada palabra será unida a la anterior y comenzará con una letra mayúscula, a excepción de la primera palabra que comenzará en minúscula. Por ejemplo: abrirPuerta, cerrarPuerta, buscarPuerta, etc:

- **Interfaz;** es un conjunto de operaciones que permiten a un objeto comportarse de cierta manera, por lo que define los requerimientos mínimos del objeto. Hace referencia a polimorfismo¹²; “también encontramos que una Interfaz representa a la comunicación entre Actores mediante un Protocolo determinado, realizando una conexión Física para realizar una función determinada tanto entre dos dispositivos como bien entre dos sistemas completamente diferentes, realizando así una comunicación.” (MasterMagazine, 2012)
- **Herencia;** se define como la reutilización de un objeto padre ya definido para poder extender la funcionalidad en un objeto hijo. Los objetos hijos heredan todas las operaciones y/o propiedades de un objeto padre.” (VARGAS M, Análisis De Sistemas, 2011).

b. Diagrama de componentes:

Representa cómo un sistema de software es dividido en componentes y muestra las dependencias entre éstos componentes. Los componentes físicos incluyen archivos, cabeceras, bibliotecas compartidas, módulos, ejecutables, o paquetes; muestra las dependencias lógicas entre componentes software, sean éstos componentes fuentes, binarios o ejecutables, se consideran en este tipo de diagramas solo tipos de componentes, instancias más específicas se encuentran en el diagrama de ejecución.

¹² **Poliformismo:** En programación orientada a objetos el polimorfismo se refiere a la posibilidad de enviar un mensaje a un grupo de objetos cuya naturaleza puede ser heterogénea. El único requisito que deben cumplir los objetos que se utilizan de manera polimórfica es saber responder al mensaje que se les envía. La apariencia del código puede ser muy diferente dependiendo del lenguaje que se utilice, más allá de las obvias diferencias sintácticas.

c. Diagrama de objetos:

Son utilizados durante el proceso de Análisis y Diseño de los sistemas informáticos en la metodología UML. Un objeto es una instancia de una clase, por lo que un diagrama de objetos puede ser visto como una instancia de un diagrama de clases. Los diagramas de objetos describen la estructura estática de un sistema en un momento particular y son usados para probar la precisión de los diagramas de clases.

d. Diagrama de paquetes:

“Los paquetes están normalmente organizados para maximizar la coherencia interna dentro de cada paquete y minimizar el acoplamiento externo entre los paquetes. Con estas líneas maestras sobre la mesa, los paquetes son buenos elementos de gestión. Cada paquete puede asignarse a un individuo o a un equipo, y las dependencias entre ellos pueden indicar el orden de desarrollo requerido” (Vélez Amelia, 2012)

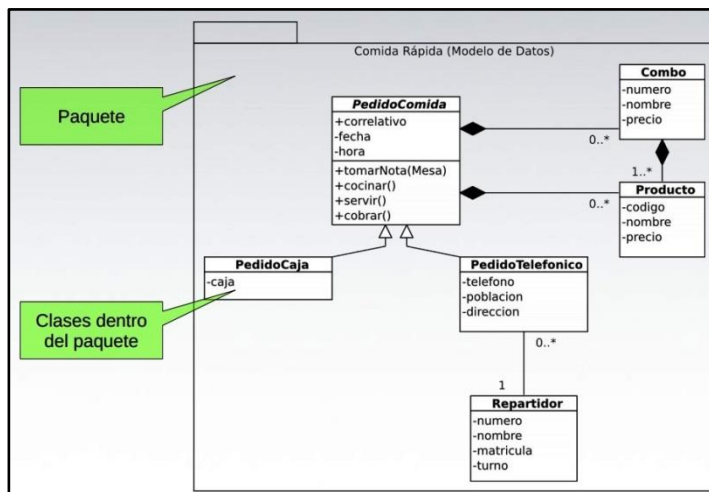


Gráfico 2: Diagrama de Paquetes.

Fuente: http://www.codecompiling.net/files/slides/UML_clase_05_UML_paquetes.pdf

Autor: Gutiérrez Damián.

e. Diagrama de actividades:

Un diagrama de actividades representa los flujos de trabajo paso a paso de negocio y operacionales de los componentes en un sistema. “En SysML el diagrama de Actividades ha sido extendido para indicar flujos entre pasos que mueven elementos físicos (e.g., gasolina) o energía (e.g., presión). Los cambios adicionales permiten al diagrama soportar mejor flujos de comportamiento y datos continuos” (CRUZ, 2012).

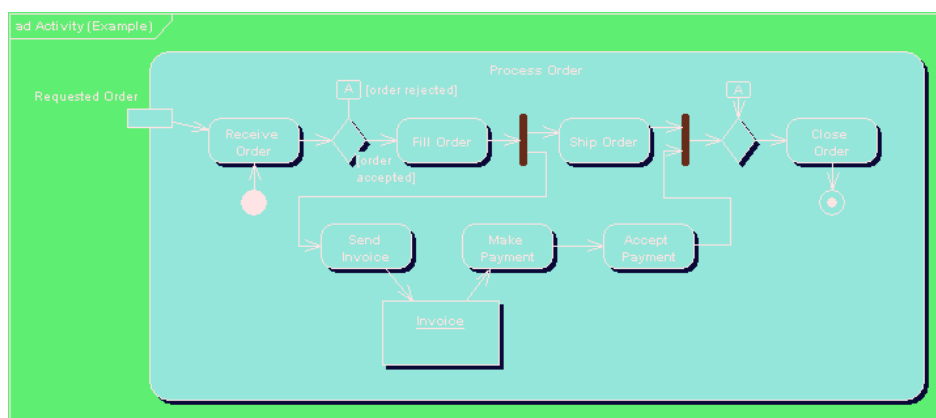


Gráfico 3. Diagrama de Actividades.

Fuente: http://www.sparxsystems.com.ar/resources/tutorial/uml2_activitydiagram.html

Autor: SparxSystem.

f. Diagrama de casos de uso:

En el Lenguaje de Modelado Unificado, es una especie de diagrama de comportamiento. Un diagrama de casos de uso es una secuencia de transacciones que son desarrolladas por un sistema en respuesta a un evento que inicia un actor sobre el propio sistema.

Los diagramas de casos de uso sirven para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y otros sistemas.

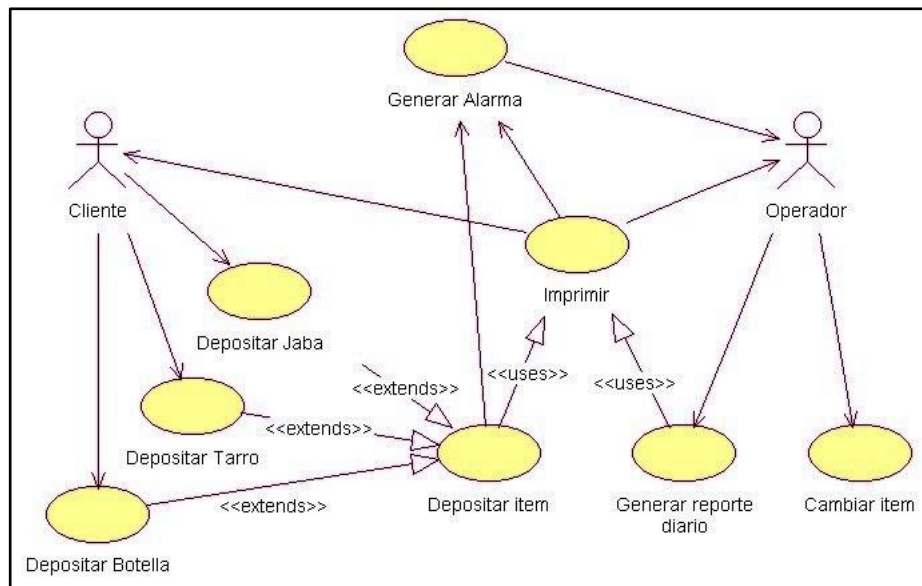


Gráfico 4. Componentes de un diagrama de casos de uso.

Fuente: <http://users.dcc.uchile.cl/~psalinas/uml/casosuso.html>

Autor: Universidad de Chile.

1.6.1.2.2 Los Diagramas de Comportamiento: Los diagramas de comportamiento se emplean para visualizar, especificar, construir y documentar los aspectos dinámicos de un sistema. Los aspectos dinámicos de un sistema de software involucran cosas tales como el flujo de mensajes a lo largo del tiempo y el movimiento físico de componentes en una red, éstos se dividen en:

- a. Diagrama de actividades;
- b. Diagrama de casos de uso; y,
- c. Diagrama de estados.

1.6.1.2.3 Los Diagramas de Interacción: son un subtipo de diagramas de comportamiento, que enfatiza sobre el flujo de control y de datos entre los elementos del sistema modelado:

a. Diagrama de secuencia

“En un diagrama de secuencia se indicarán los módulos o clases que forman parte del programa y las llamadas que se hacen en cada uno de ellos para realizar una tarea determinada. El detalle del diagrama depende de la fase en la que estemos, lo que pretendamos contar con el diagrama y a quién. En una primera fase de diseño

podemos poner clases grandes y ficticias, que representen un paquete/librería o, si nuestro programa está compuesto por varios ejecutables corriendo a la vez, incluso clases que representen un ejecutable.” (Cáceres Tello, 2012, pág. 1)

1.6.1.3 Proceso Unificado Racional

Se utilizará el Proceso Unificado Racional ya que contiene los ciclos de vida que se adaptan al proceso del sistema, además, permite llevar un esquema ordenado de desarrollo.

“El Proceso Unificado Racional (*Rational Unified Process* en inglés, habitualmente resumido como RUP) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

También se conoce por este nombre al software desarrollado por Rational, hoy propiedad de IBM, el cual incluye información entrelazada de diversos artefactos y descripciones de las diversas actividades. Está incluido en el Rational Method Composer (RMC), que permite la personalización de acuerdo a necesidades” (MAGAÑA, 2011).

1.6.1.4 Principios de desarrollo

El RUP está basado en 6 principios clave que son los siguientes:

- a. **Adaptar el proceso:** Este deberá adecuarse a las necesidades del cliente debido a la importancia de la interacción con él. Las características propias del proyecto u organización. El tamaño del mismo, así como su tipo o las regulaciones que lo condicionen, influirán en su diseño específico;
- b. **Equilibrar prioridades:** Los requisitos de los diversos participantes pueden ser diferentes, contradictorios o disputarse recursos limitados. *Debe encontrarse un equilibrio que satisfaga los deseos de todos.* Gracias a este equilibrio se podrán corregir desacuerdos que surjan en el futuro;
- c. **Demostrar valor iterativamente:** Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas¹³. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados;
- d. **Colaboración entre equipos:** El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requisitos, desarrollo, evaluaciones, planes, resultados, etc.
- e. **Elevar el nivel de abstracción:** Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o marcos de referencia (frameworks¹⁴) por nombrar algunos. Un alto nivel de abstracción permite discusiones sobre diversos niveles y soluciones arquitectónicas. Éstas se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con el lenguaje UML; y,
- f. **Enfocarse en la calidad:** El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción. El

¹³ **ETAPAS ITERADAS:** En matemáticas, es una función que es compuesta consigo misma, en forma repetida, en un proceso llamado iteración. Las funciones iteradas son objeto de profundos estudios en el campo de los fractales y sistemas dinámicos.

¹⁴ **FRAMEWORKS:** En el desarrollo de software, un framework o infraestructura digital, es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

aseguramiento de la calidad forma parte del proceso de desarrollo y no de un grupo independiente.

1.6.1.4.1 Ciclo De Vida

El ciclo de vida RUP es una implementación del desarrollo en espiral, esto permite organizar las tareas en fases e iteraciones.

RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades. En la figura muestra cómo varía el esfuerzo asociado a las disciplinas según la fase en la que se encuentre el proyecto RUP.

Las primeras iteraciones (en las fases de Inicio y Elaboración) se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una baseline (Línea Base) de la arquitectura. Durante la fase de inicio las iteraciones hacen mayor énfasis en actividades de modelado del negocio y de requisitos.

En la fase de elaboración, las iteraciones se orientan al desarrollo de la baseline de la arquitectura, abarcan más los flujos de trabajo de requisitos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la baseline de la arquitectura.

En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones.

Para cada iteración se selecciona algunos Casos de Uso, se refina su análisis y diseño y se procede a su implementación y pruebas. Se realiza una pequeña cascada para cada ciclo. Se realizan tantas iteraciones hasta que se termine la implementación de la nueva versión del producto.

En la fase de transición se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

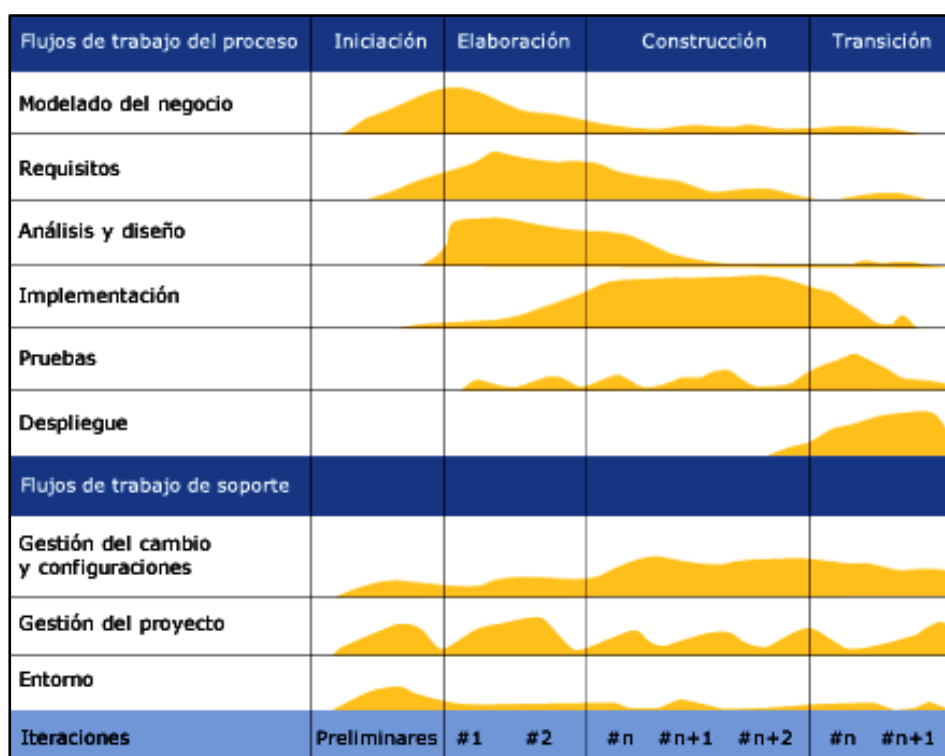


Gráfico 5. Esfuerzo en actividades según fase del proyecto.

Fuente: http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational.

Autor: Wikipedia.org

1.6.1.5 Project Management Professional, PMP

Se utiliza Project Management Professional, ya que permitirá llevar cronogramas de trabajo exactos para el desarrollo del sistema

La administración del proyecto se desarrollará mediante el respaldo de **PMP(Profesionalismo en la Administración de Proyectos o Project Management Professional)**, la cual es una Guía del Conjunto de Conocimientos de la Administración de Proyectos denominada PMBOK¹⁵, esta ciencia es la aplicación de conocimientos, métodos, habilidades, herramientas, técnicas y requerimientos abordando planeación, administración, evaluación de riesgos y toma de decisiones con una actitud fuertemente orientada a resultados, objetivos claros, actividades programadas y trabajo de equipo.

1.7 Procesos de negocio

Se especifica 3 procesos que se manejarán en el sistema:

1.7.1 Cliente: El cliente podrá visualizar la información de su embarque en la web, a través de la url, que se le proporcionará; la cual se visualizará por 3 tipos de opciones:

- Visualización del último embarque;
- Visualización del embarque por fecha; y,
- Visualización del embarque por búsqueda.

Cada uno de los embarques contiene la información interna del mismo, que el cliente necesita, estos son los documentos necesarios para la exportación y desaduanización de la carga, como certificados, facturas, guías. Esta información podrá ser visualizada de 2 maneras:

- Mediante descarga; y,
- Mediante visualización en el browser en formato pdf.

¹⁵ **PMBOK:** es el conjunto de conocimientos en Dirección/Gestión/Administración de Proyectos generalmente reconocidos como "buenas practicas", y que se constituye como estándar de Administración de proyectos. La Guía PMBOK® comprende dos grandes secciones, la primera sobre los procesos y contextos de un proyecto, la segunda sobre las áreas de conocimientos específicos para la gestión de un proyecto.

1.7.2 Bróker: El bróker es el agente intermedio entre la agencia de carga y el cliente, el mismo que se encarga de desaduanizar la carga en los aeropuertos. El bróker podrá visualizar la información de su embarque en la web, a través de la url, que se le proporcionará; dicha información será visualizada por 3 tipos de opciones:

- Visualización del último embarque;
- Visualización del embarque por fecha; y
- Visualización del embarque por búsqueda.

La información que visualiza el bróker es solamente la necesaria para llevar a cabo su labor, la misma se visualizará de 2 maneras:

- Mediante descarga; y
- Mediante visualización en el browser en formato pdf.

1.7.3 Florícola: El cliente podrá visualizar la información de su embarque en la web, a través de la url que se le proporcionará, Cada opción de visualización contendrá la información de fechas de vuelo, con los números de guía y cantidad de cajas que vuelan por cada embarque; dicha información será visualizada por 3 tipos de opciones:

- Visualización del último embarque;
- Visualización del embarque por fecha; y,
- Visualización del embarque por búsqueda.

Esta información podrá ser visualizada de 2 maneras:

- Mediante descarga; y,
- Mediante visualización en el browser en formato pdf.

CAPÍTULO II.

ANÁLISIS

2.1 PMP (profesionalismo en la administración de proyectos. Project Management Professional)

La administración del proyecto se desarrolla mediante el respaldo de PMP, la cual es una Guía del Conjunto de Conocimientos de la Administración de Proyectos denominada PMBOK.

Project Management Professional (PMP)¹⁶ es una certificación (credencial) ofrecida por el Project Management Institute (PMI)¹⁷.

El examen de PMP (Project Management Professional) se basa en la especificación de indagación PMP (*PMP Examination Specification*), que se describe por seis dominios:

1. Inicio, (11%);
2. Planificación, (23%);
3. Ejecución, (27%);
4. Seguimiento y control, (21%);
5. Cierre, (9%); y,
6. Responsabilidad social y profesional (9%).

¹⁶ **Project Management Professional (PMP)** es una certificación (credencial) ofrecida por el Project Management Institute (PMI). La credencial se obtiene mediante la documentación de 3 a 5 años de experiencia en gestión de proyectos, completar 35 horas de formación relacionadas con la gestión de proyectos, y obteniendo un determinado porcentaje de las preguntas en un examen escrito de opción múltiple.

¹⁷ **El Project Management Institute (PMI®)** es una organización internacional sin fines de lucro que asocia a profesionales relacionados con la Gestión de Proyectos. Desde principios de 2011, es la más grande del mundo en su rubro, dado que se encuentra integrada por más de 380.000 miembros en cerca de 170 países. La oficina central se encuentra en la localidad de Newtown Square, en la periferia de la ciudad de Filadelfia, en Pennsylvania (Estados Unidos).

2.1.1 PMBOK

La Guía de los fundamentos de la dirección de proyectos (más conocida como PMBOK) es el estándar más ampliamente reconocido para manejar y administrar proyectos.

“Finalidad y Carácter del PMBOK: es la agrupación de conocimientos para dirigir un proyecto, los mismos que representan un conjunto vivo, extraordinariamente amplio, producto tanto de la experiencia como del estudio y del desarrollo sistemáticos. Este conjunto de conocimientos se encuentra distribuido en miles de personas, organizaciones y textos; la finalidad del PMBOK es la de identificar el subconjunto de buenas prácticas” (OLADE, 2011).

Todo proyecto (así como sus distintas fases e interacciones) tiene que transitar por una serie de actividades bajo el gobierno de un grupo de procesos más general de supervisión y cierre, como son:

- Inicio;
- Planeación;
- Ejecución; y,
- Cierre.



Gráfico 6. Proceso PMBOK

Autor: Sánchez Gallardo José.

Éstos grupos de procesos no representan fases, sino que, equivalen al modelo “planear, hacer, revisar y actuar”:

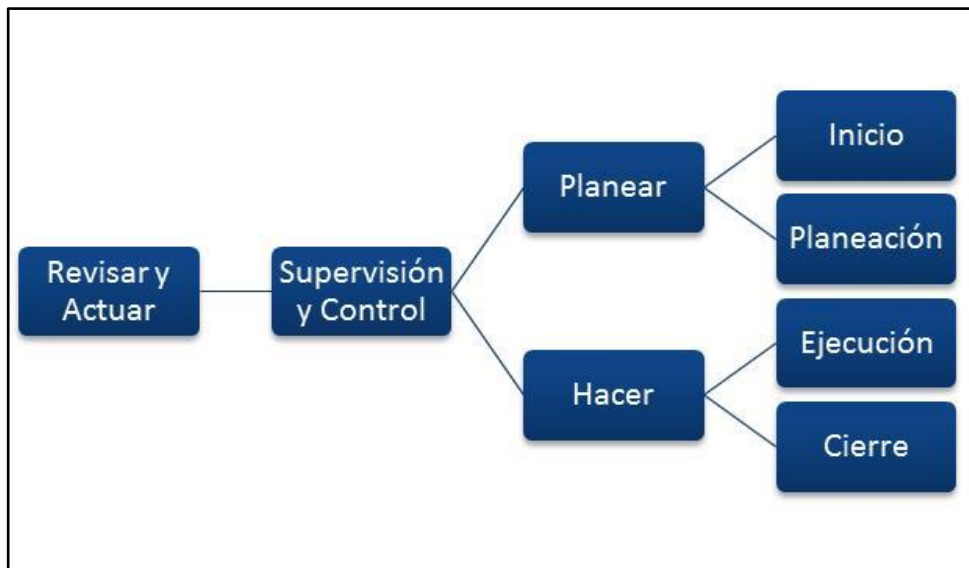


Gráfico 7. Modelo Equivalente PMBOK.

Autor: Sánchez Gallardo José.

PMBOK está representado en las nueve (9) áreas de conocimiento, y que son propiamente las que contienen las técnicas para poder realizar los proyectos:

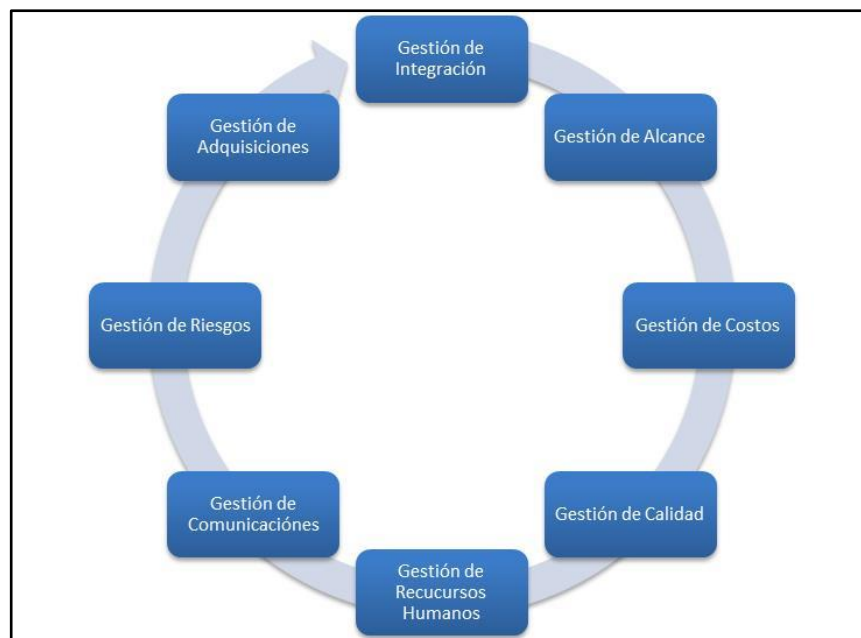


Gráfico 8. Áreas del Conocimiento.

Autor: Sánchez Gallardo José.

2.1.1.1 Estructura del PMBOK:

El PMBOK® es un compendio de mejores prácticas, agrupadas de cierta manera, heredadas de diversas industrias y disciplinas que conforman un modelo metodológico. El PMBOK® en sí no es una metodología que “deba” ser seguida al pie de la letra, es sólo una guía, completa y elaborada, de lo que normalmente un gerente de proyectos debe llevar a cabo, explicado en un buen nivel de detalle y separando procesos que normalmente se llevan a cabo de forma simultánea.

- *Estructura*

El PMBOK® establece la administración de proyectos como un conjunto de nueve áreas de conocimiento que deben ser dominadas por el Project Manager y contiene una serie de procesos correspondientes a los pasos necesarios para ser completamente cubiertas. Cada proceso establece unas entradas (documentos), técnicas (mejores prácticas) y salidas (nuevamente documentos). Tanto las entradas como las salidas conectan a los diferentes procesos entre sí para formar una completa red sobre la que se puede establecer una metodología.

El PMBOK® puede verse de dos formas diferentes, cual si fuera una matriz que puede leerse por columnas o filas. La forma estándar como está estructurado el documento establece áreas de conocimiento. La forma útil para el gerente de proyectos y la organización es, sin embargo, por grupos de procesos de Inicio, Planeación, Ejecución, Control y Cierre.

- *Las áreas de conocimiento definidas en el PMBOK® son:*

- ▶ Gestión de Integración: Procesos requeridos para integrar todas las actividades, documentos y recursos del proyecto;

- ▶ Gestión de Alcance: Procesos requeridos para identificar todo el trabajo requerido y sólo éste para obtener los entregables del proyecto y cumplir los objetivos;
- ▶ Gestión de Tiempo: Procesos requeridos para asegurar que el proyecto es finalizado a tiempo;
- ▶ Gestión de Costos: Procesos requeridos para asegurar que el proyecto es finalizado dentro de un presupuesto aprobado;
- ▶ Gestión de Calidad: Procesos requeridos para asegurar que el proyecto cumple los requerimientos y necesidades por los cuales fue emprendido;
- ▶ Gestión de Comunicaciones: Procesos requeridos para asegurar la generación, distribución, almacenamiento y disposición última de toda la información del proyecto, a tiempo y de forma adecuada;
- ▶ Gestión de Recursos Humanos: Procesos requeridos para administrar eficientemente la gente que participa en el proyecto;
- ▶ Gestión de Riesgos: Procesos requeridos para identificar, analizar y responder efectivamente a los riesgos del proyecto; y,
- ▶ Gestión de Adquisiciones: Procesos requeridos para adquirir bienes y servicios fuera de la organización del proyecto.

Cada área de conocimiento incluye varios procesos que se presentan en la siguiente tabla:

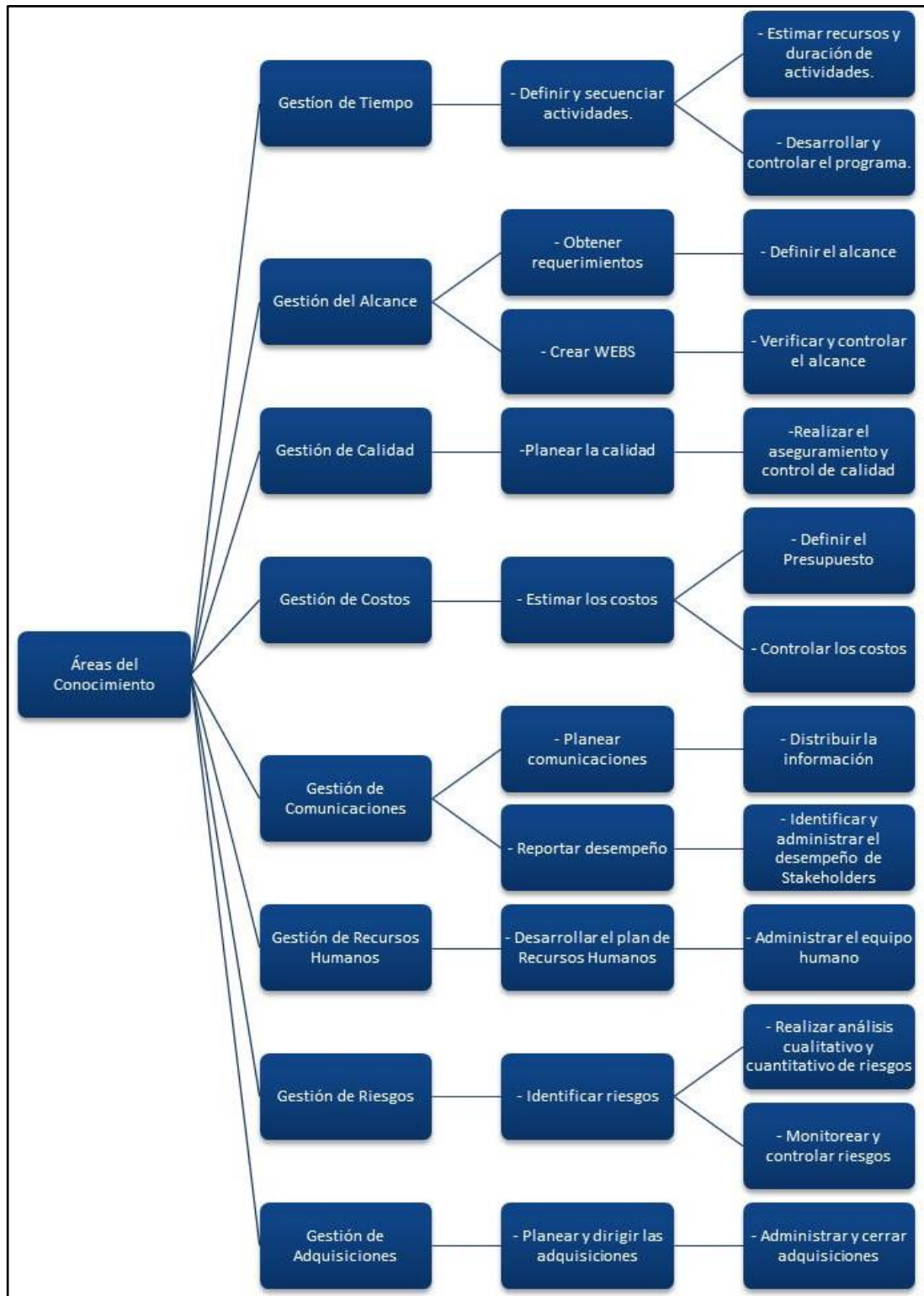


Gráfico 9. Áreas de Conocimiento

Autor: Sánchez Gallardo José.

- *Representación Por Grupos De Proceso:*

Debido a la aparente desconexión entre procesos y áreas, el PMBOK® también define una estructura por grupos de procesos. Estos grupos son simplemente la secuencia lógica que sigue cualquier proyecto: Inicio, Planeación, Ejecución, Control y Cierre.

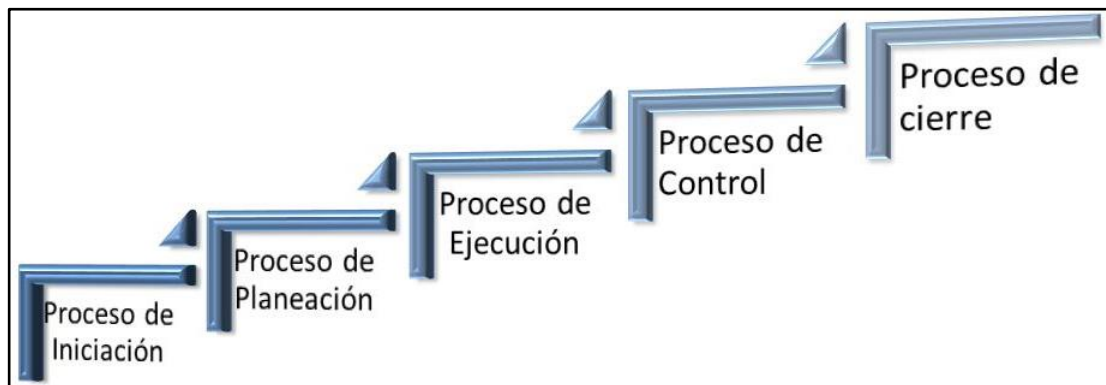


Gráfico 10. Representación de Grupos de Procesos en PMBoK.

Autor: Sánchez Gallardo José.

En esta representación el énfasis se encuentra en las interrelaciones de los grupos de procesos, en donde se evidencia un ciclo permanente entre planeación, ejecución y control que claramente indica que la planeación debe ser modificada de acuerdo a la situación del proyecto en un momento particular.

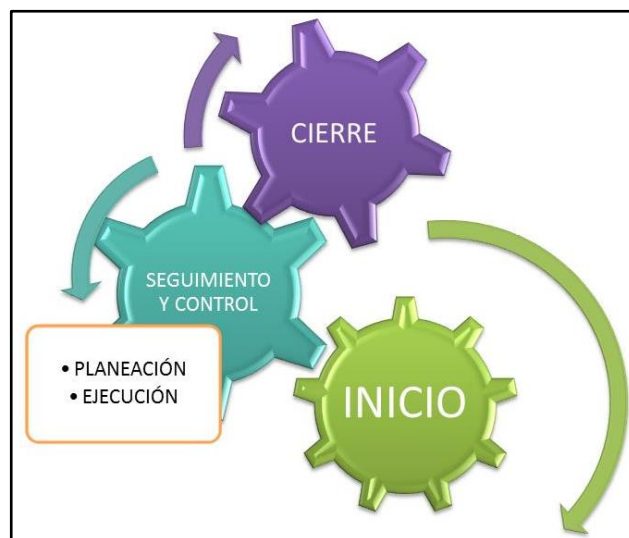


Gráfico 11. Representación de Grupos de Procesos en PMBoK Internos.

Autor: Sánchez Gallardo José.

A continuación se presentan los procesos subyacentes dentro de los grupos de proceso:

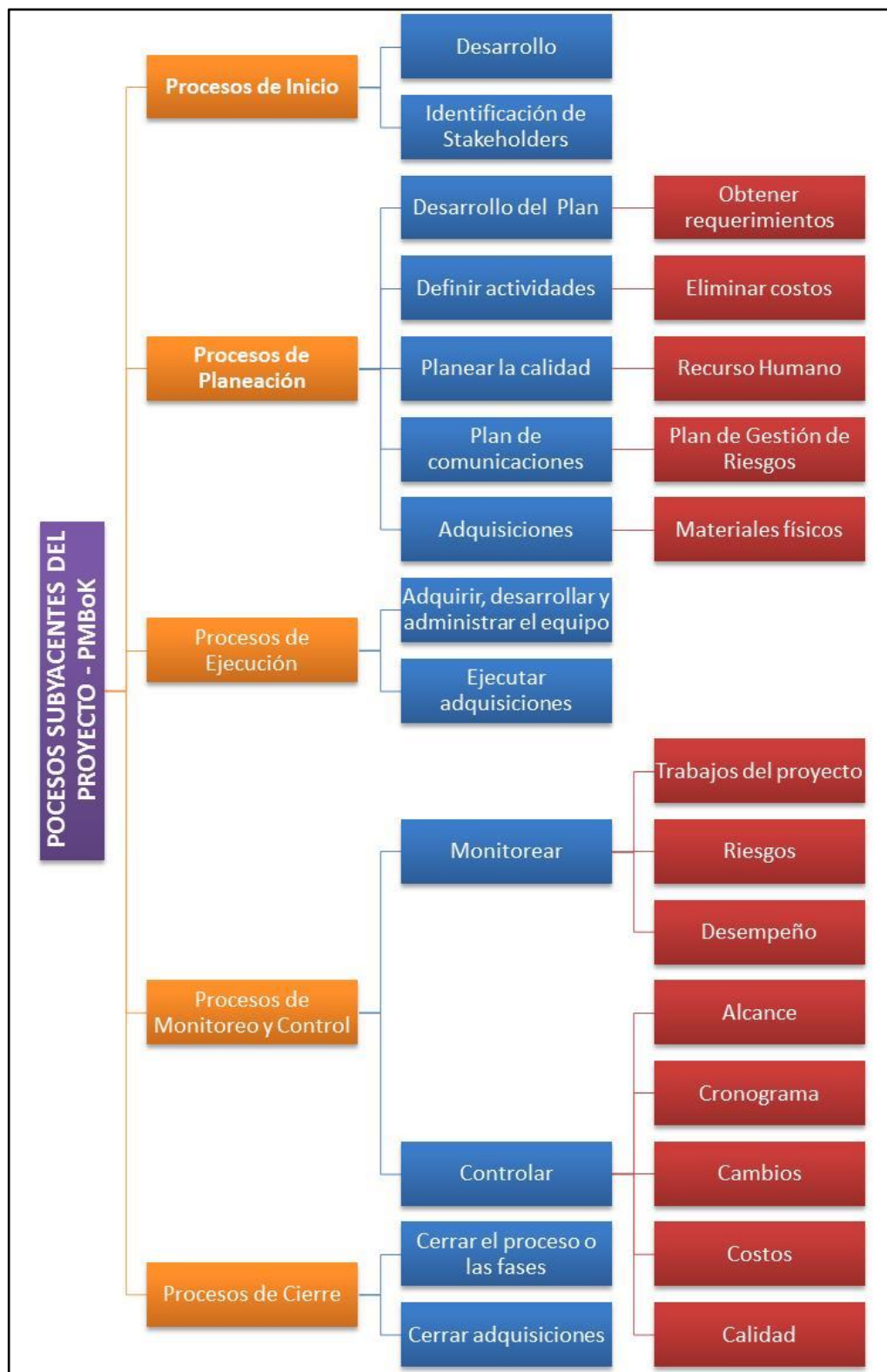


Gráfico 12. Procesos de Planeación del Proyecto según el PMBoK.

Autor: Sánchez Gallardo José.

2.2 Diagramas BPMN (notación para el modelado de procesos de negocio)

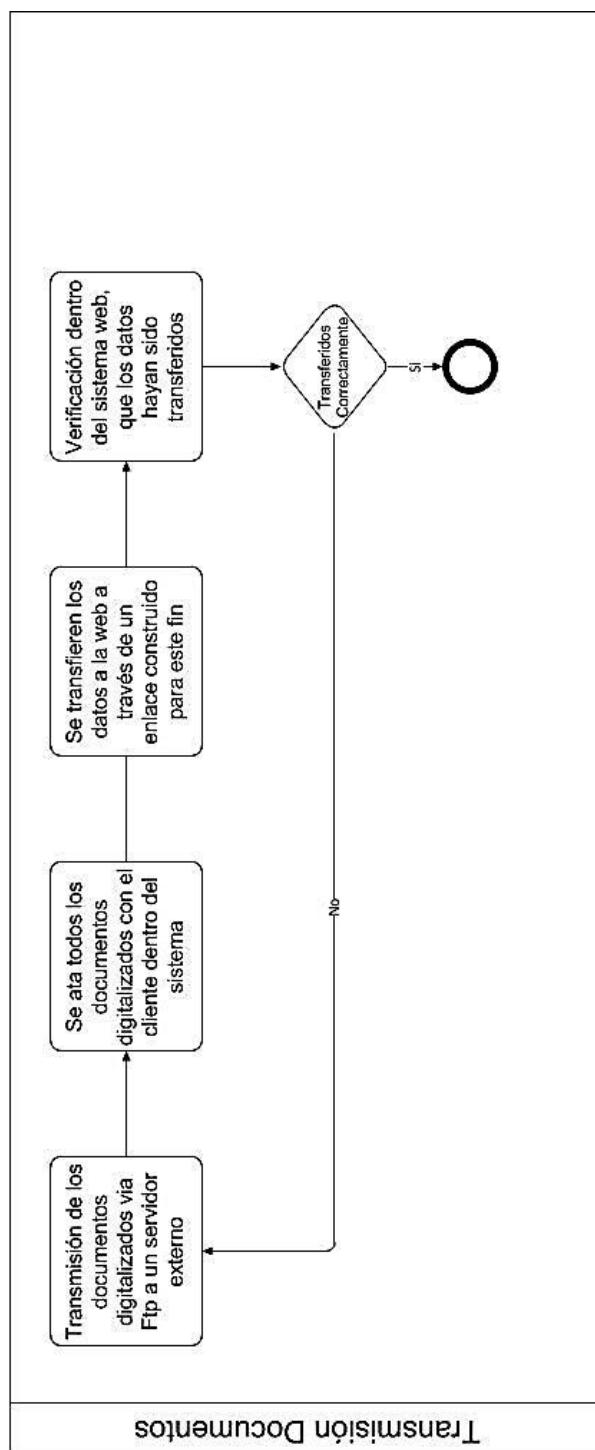


Gráfico 13. Diagrama BPMN Transmisión de Documentos.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

2.2.1 Especificación de requerimientos funcionales

2.2.1.1 Módulo Para Clientes

A este módulo solamente ingresan los clientes que tengan un usuario de ingreso y contraseña de ingreso.

- *Procesos:*

Registro de Reservación carga: Despliega un listado completo de las reservaciones que tiene el cliente, la consulta es por rango de fechas. Rastreo de Traking de embarques que tiene el cliente, la consulta es por:

- Último embarque;
- Embarques por fecha; y,
- Número de guía aérea.

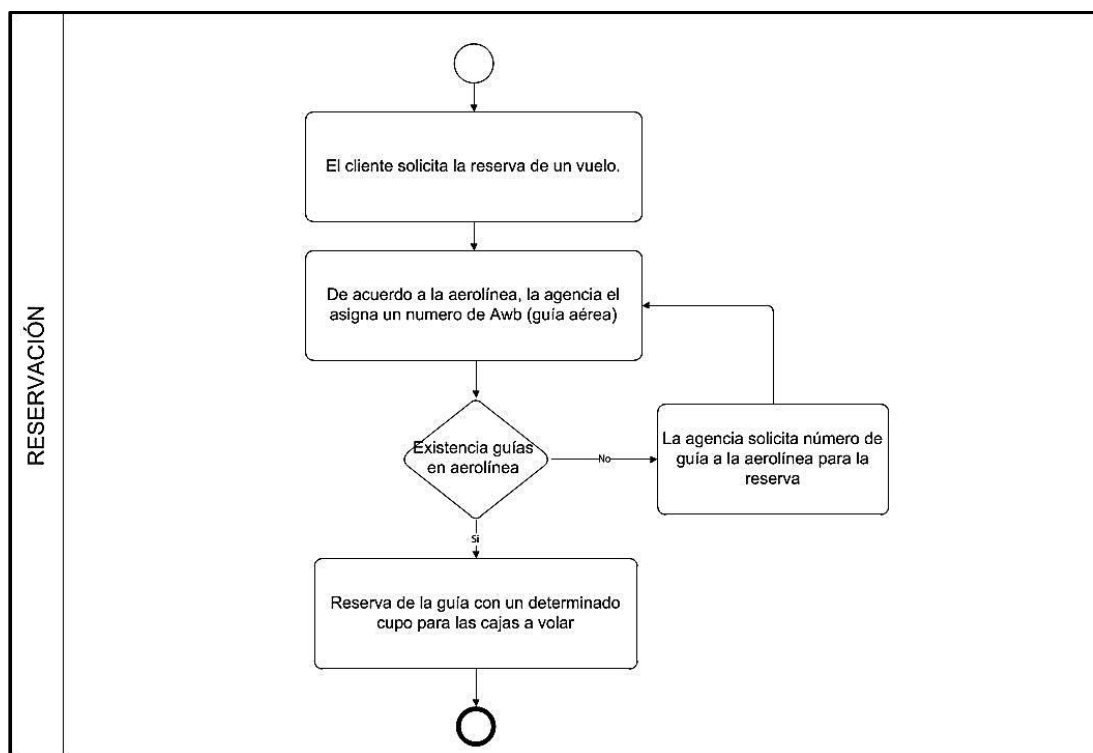


Gráfico 14. Diagrama: Reservación.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

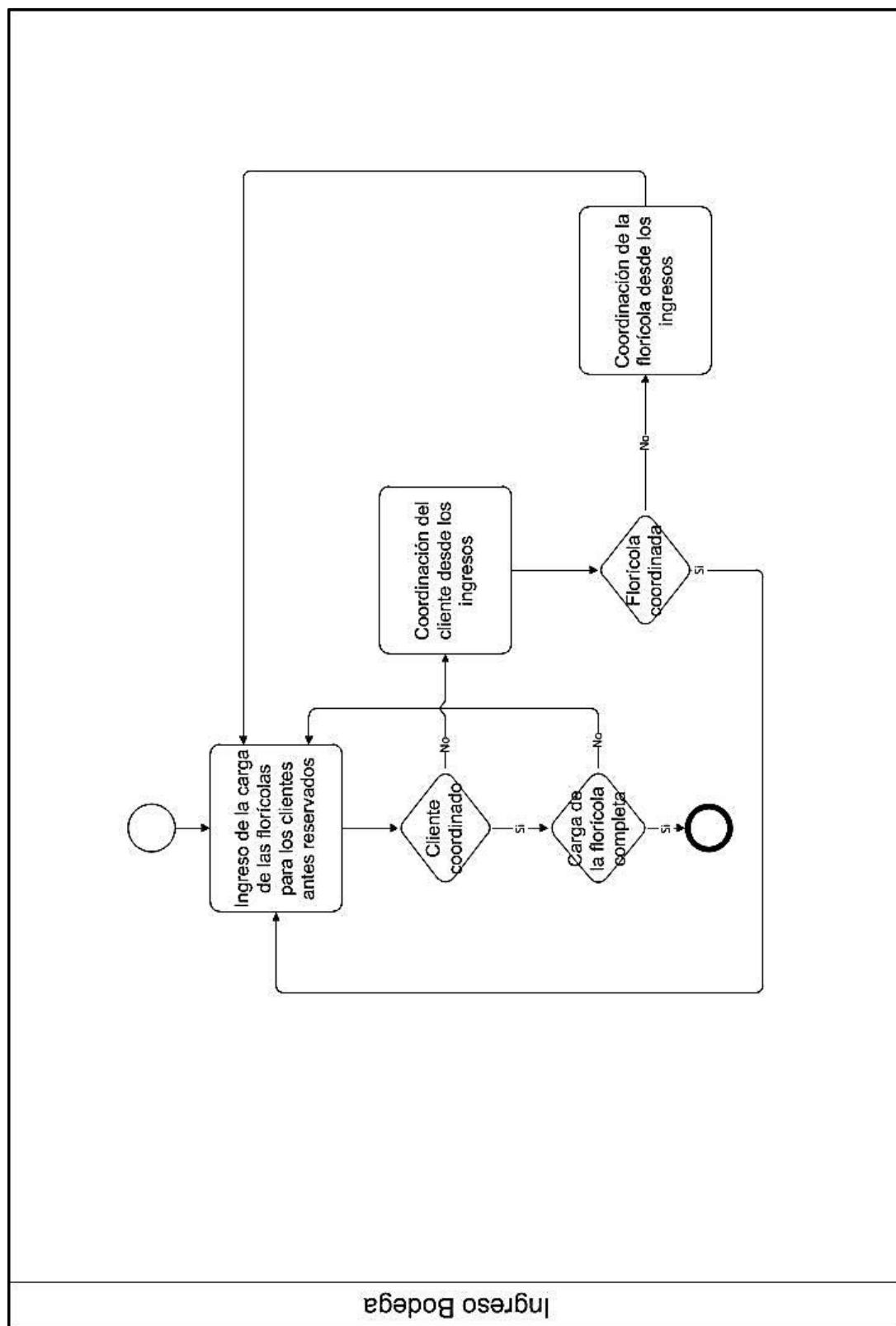


Gráfico 15. Diagrama: Ingreso Bodega.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José

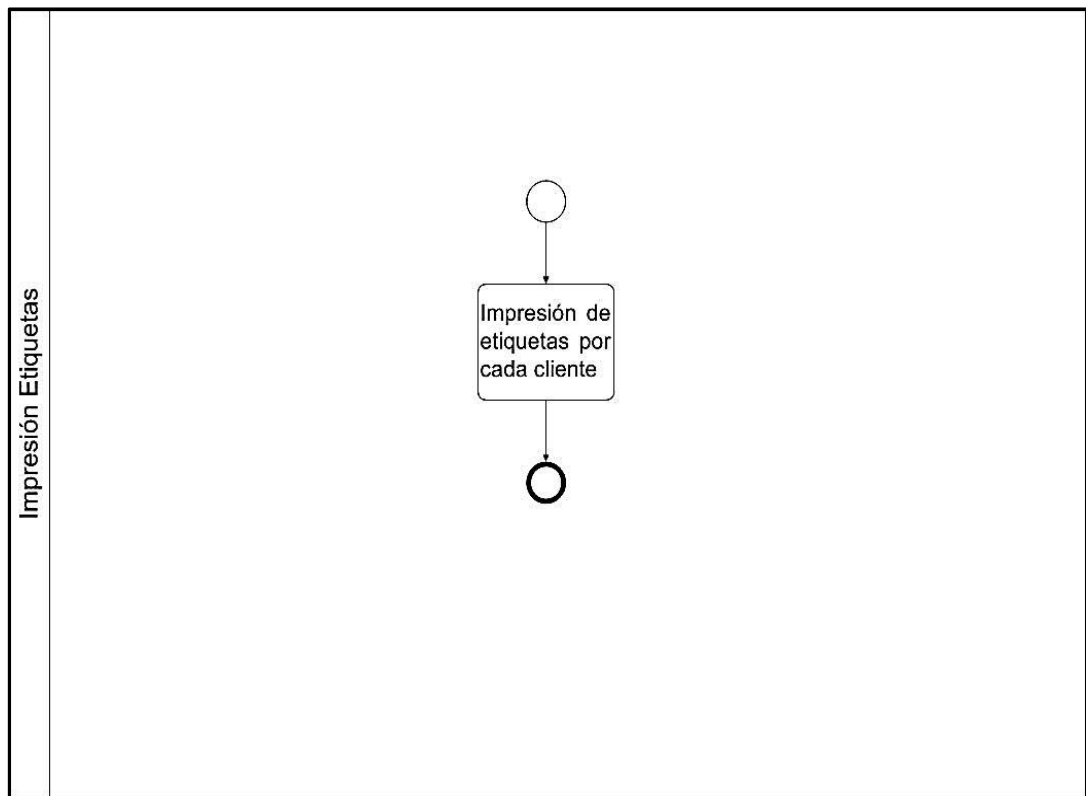


Gráfico 16. Diagrama: Etiquetas.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

Despliega documentos de exportación anexados al embarque, las opciones son:

- Último embarque;
- Por número de guía aérea; y,
- Por fecha de embarque.

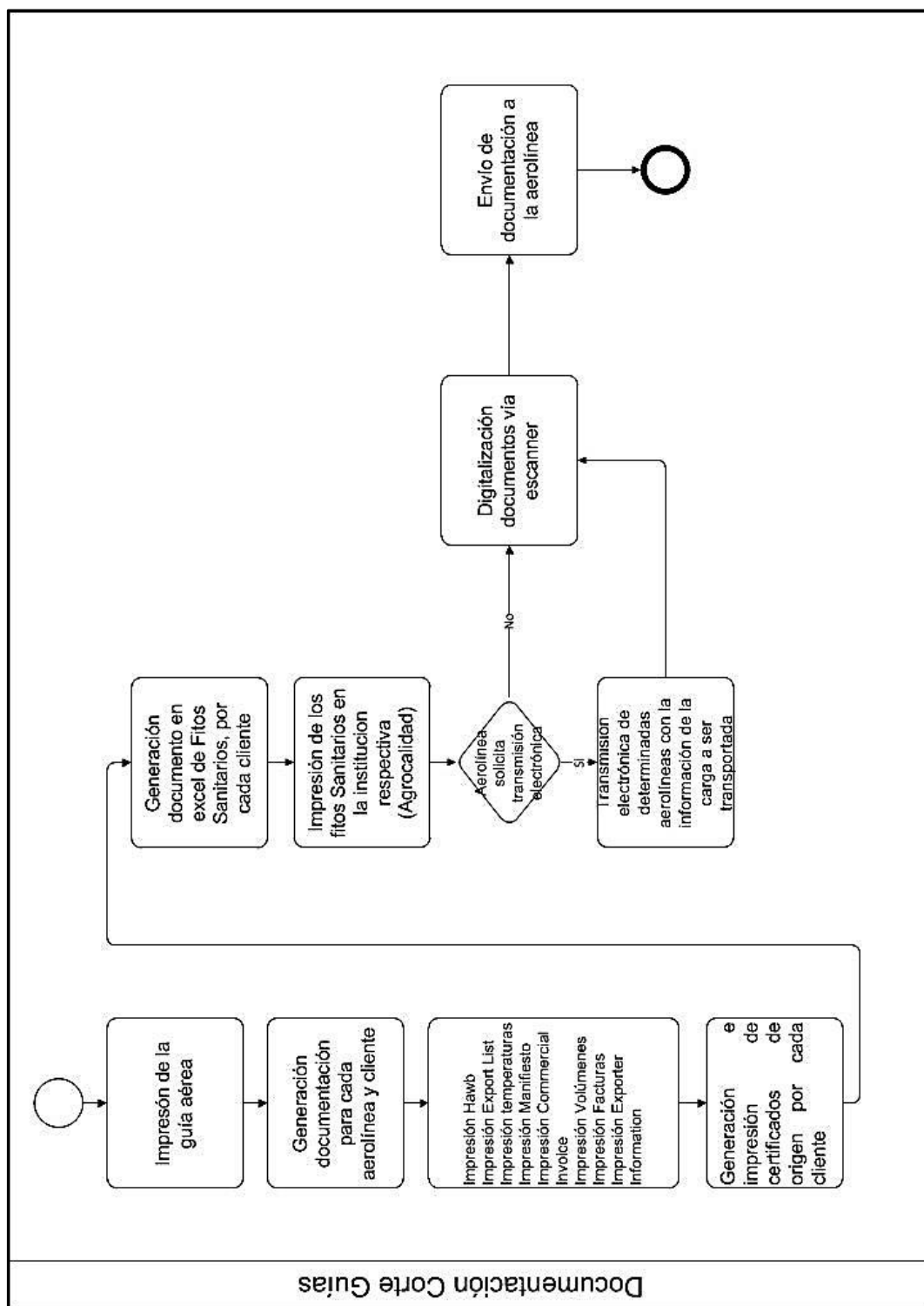


Gráfico 17. Diagrama: Documentación.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

Monitoreo de carga: permite llevar un rastreo del vuelo post salida, por cada uno de los destinos, las opciones son:

- Último embarque; y,
- Número de guía aérea.

En todas las opciones del cliente tiene un rastreo de la carga que depende de la conexión a Internet del cual disponga, y le presenta al cliente todo el proceso de embarque, desde la reservación, coordinación, ingreso bodega, corte de guía aérea, documentos de exportación, etc.

- Adicionales solicitados por el CLIENTE:

- Visualizar listado general de guías aéreas (fecha, numero de guía, consigne, número de cajas, estado de vuelo);
- Visualizar los siguientes datos:
 - Carrier, Bróker, número de guía, tipo de pago, aeropuerto de destino;
 - Peso, Flete neto, valor aerolínea, valor agencia, total costo, número de cajas, tarifa;
 - Detalle de ruta, con fecha y hora de vuelos; y,
 - Estado, shipper, hawb, # FUE, Número de cajas (reservadas / voladas), fecha, temperatura.
- Visualizar los siguientes documentos:
 - AWB;
 - Export List;
 - Fitos Sanitarios; y,
 - Certificado de origen.
 -
- El módulo WEB de clientes estará sujeto a cambios, definidos en la etapa de análisis.

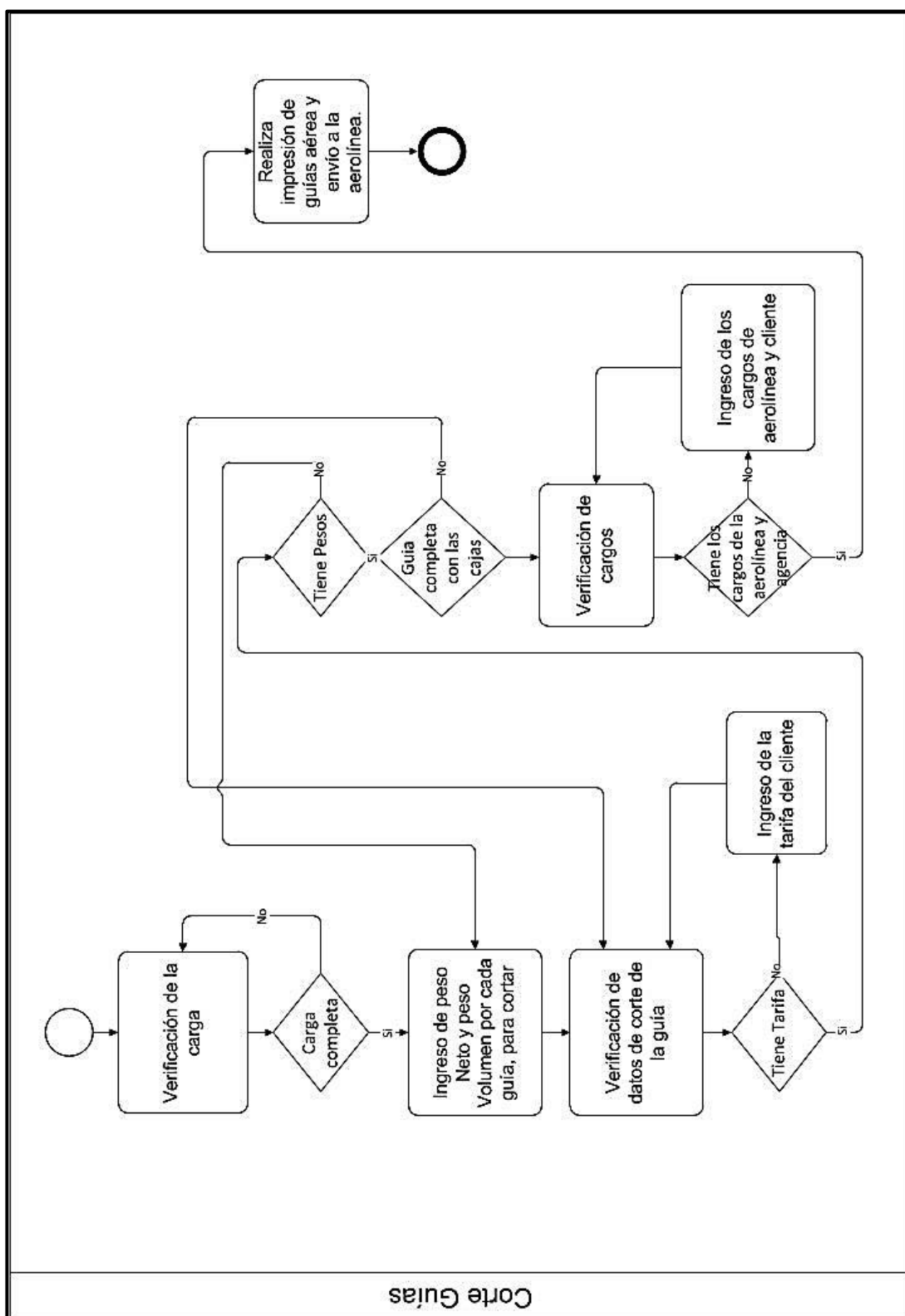


Gráfico 18. Diagrama: Corte de Guías.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

2.2.1.2 Módulo de Bróker

Este módulo es para los Brókers, la información que presenta es solamente la cabecera de la guía aérea, no el detalle ya que solamente necesitan datos puntuales.

- ***Procesos:***

- Rastreo de embarques, se encuentra detallada la cabecera de la guía aérea por: último embarque, fecha de embarque, número de guía aérea;
- Detalla los documentos de exportación a los cuales el cliente permite que el Bróker tenga acceso; y,
- Monitoreo de carga, donde se realiza un rastreo del vuelo por cada destino, confirmando la llegada o demora al mismo, por: último embarque, fecha de embarque, número de guía aérea.

- Adicionales solicitados por el CLIENTE:

- El módulo WEB de Brókers estará sujeto a cambios, mismo que serán definidos en la etapa de análisis.

2.2.1.3 Módulo De Florícolas

A éste módulo tienen acceso las florícolas autorizadas y poseen clave de acceso al sistema.

- ***Procesos:***

- *La florícola realiza coordinación*, éstos no podrán ver todos los clientes por razones de seguridad. La agencia de carga mediante una opción

podrá autorizar a la florícola a coordinar, así podrá abrir el sistema a todas las florícolas o a una florícola específicas;

- En la *Visualización del Detalle de Embarque* se despliegan los datos de la guía aérea, las opciones son: Último embarque, Fecha de embarque, Número de guía aérea;
- *Registro de Guías certificadas generadas*, se obtiene tanto los recibos semanales, como las facturas mensuales. Así la florícola podrá tener un estado de cuenta completo de las guías certificadas que la agencia entrega;
- *Registrar FITO sanitario vigente*, se anexa una imagen digitalizada del mismo; y,
- *Monitoreo de embarque* que puede ver su carga por: Último embarque, Fecha de embarque, Número de guía aérea.

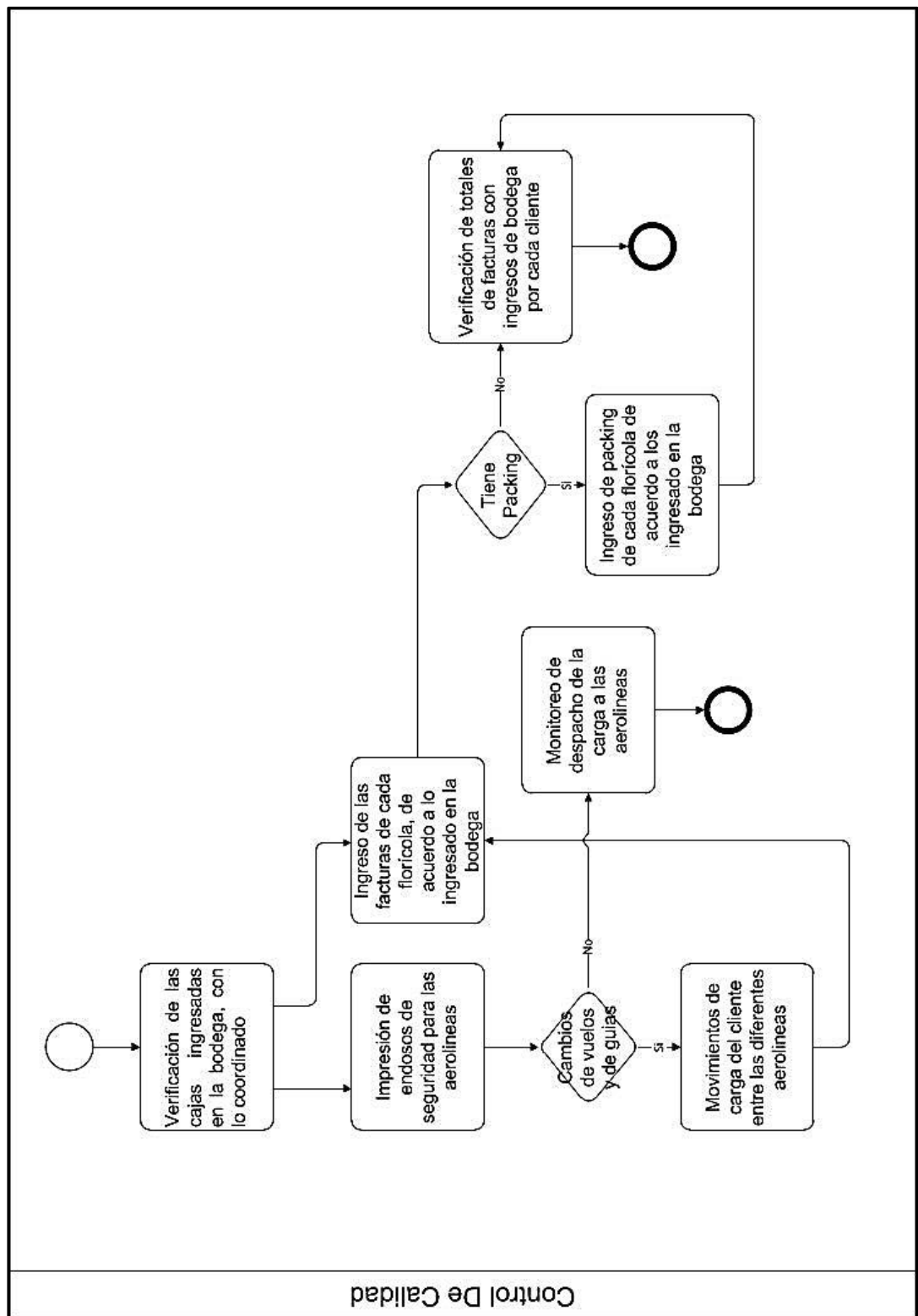


Gráfico 19. Diagrama: Control de Calidad.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- Adicionales solicitados por el CLIENTE:
 - El módulo WEB de fincas estará sujeto a cambios, definidos en la etapa de análisis;
 - A las fincas se les permite realizar la coordinación, tomando en cuenta que se podrá realizar la coordinación de una en una, o la opción de coordinar mediante carga de archivos;
 - Visualizar los siguientes datos:
 - Datos generales de la finca, nombre, dirección, teléfono, país, etc;
 - Carrier, Bróker, número de guía, guía master, tipo de pago, aeropuerto de destino;
 - Peso, Flete neto, deuda, total costo, número de cajas, tarifa;
 - Detalles de la operación, hora y fecha de vuelo, delivery, llegada;
 - Detalle de ruta, con fecha y hora de vuelos;
 - Estado, shipper, hawb, # FUE, Número de cajas (reservadas/ voladas), fecha, temperatura; y,
 - Coordinar las cajas a volarse para una determinada fecha y cliente. (campos a definir con reservas).
 - Visualizar los siguientes documentos:
 - AWB;
 - Export List;
 - Fitos Sanitarios; y
 - Certificado de origen.

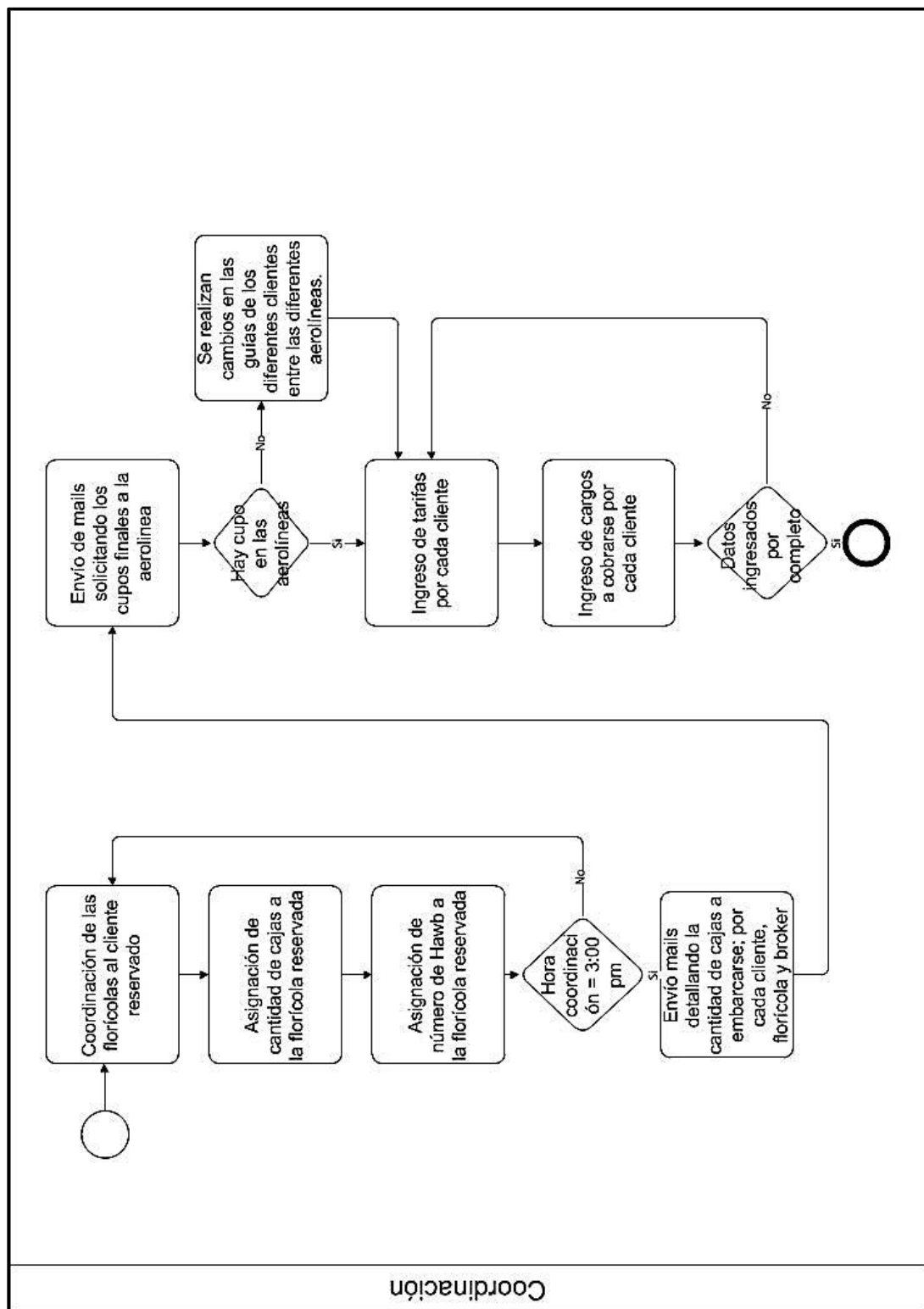


Gráfico 20. Diagrama: Coordinación.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

2.3 Diagramas UML

2.3.1 Diagrama de casos de uso

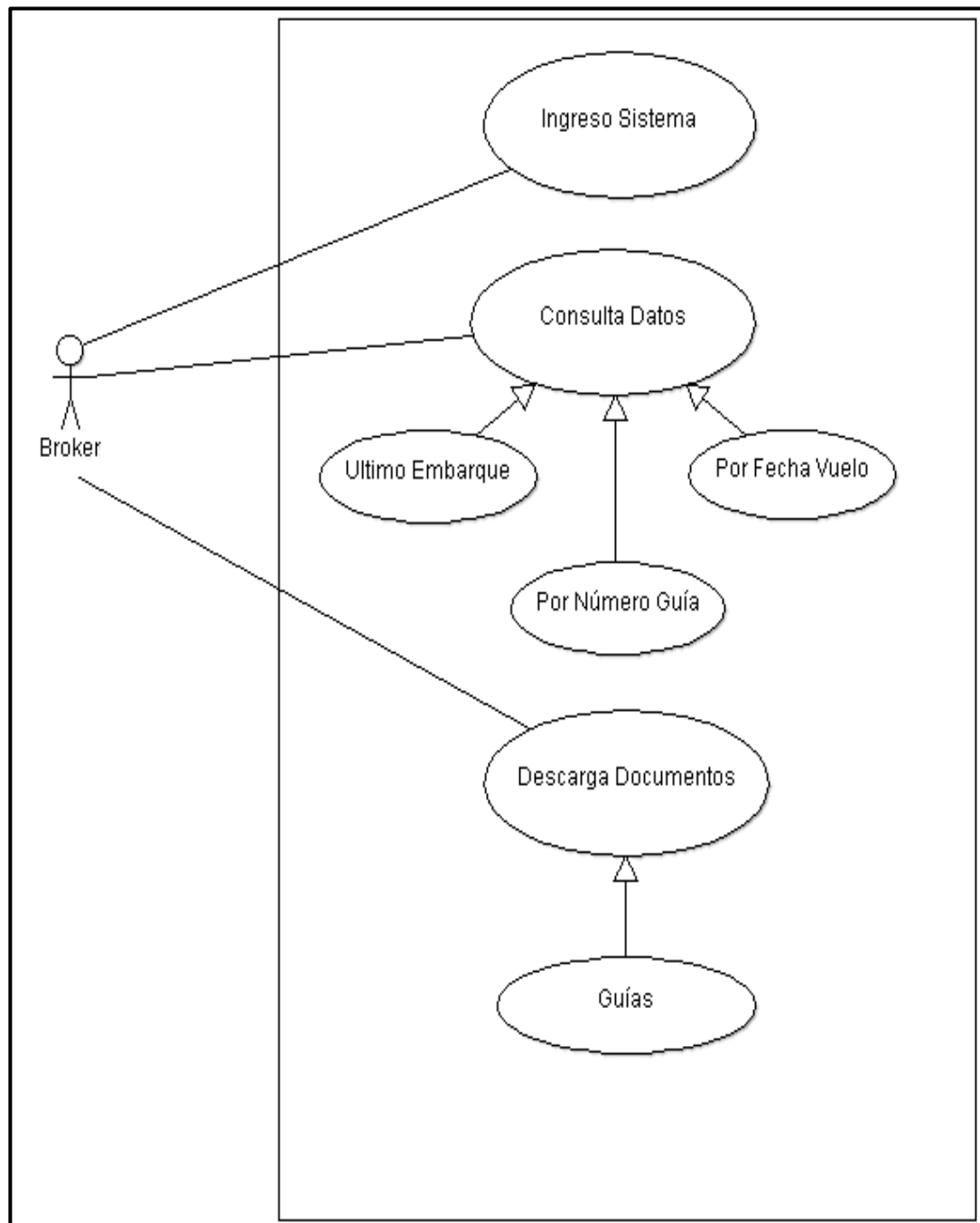


Gráfico 21. Diagrama en Caso de Uso – Bróker.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

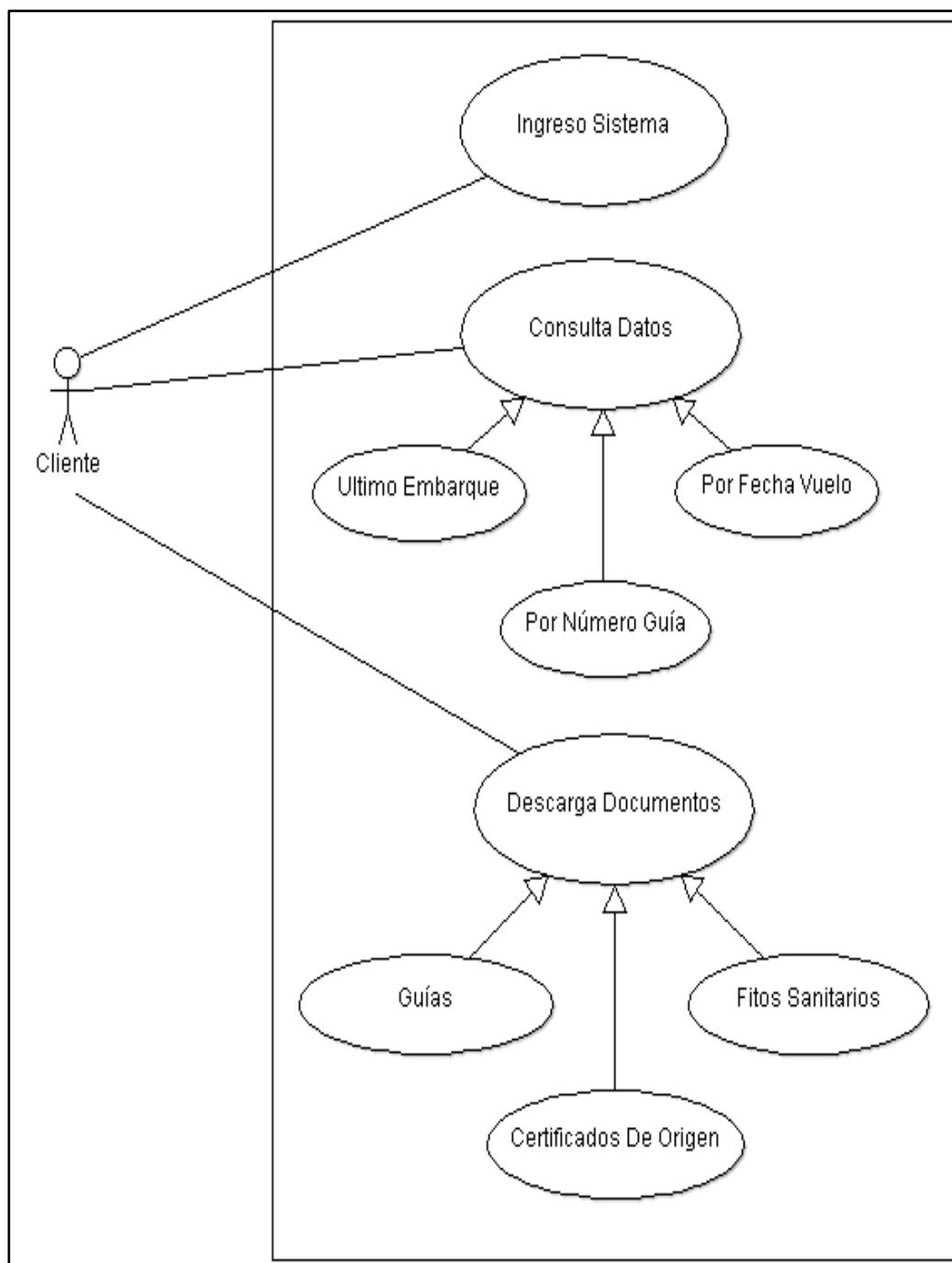


Gráfico 22. Diagrama en Caso de Uso – Cliente.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

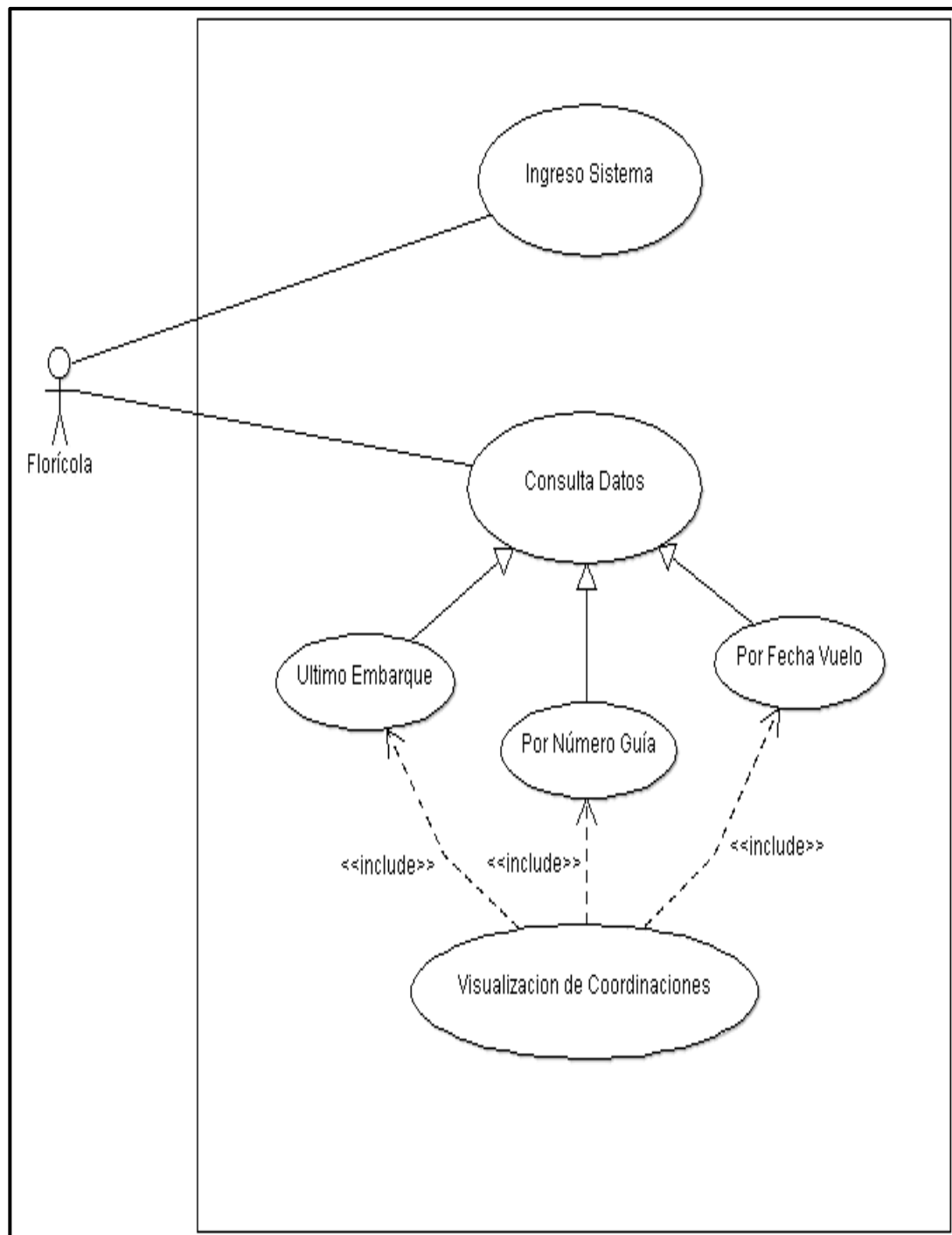


Gráfico 23. Diagrama en Caso de Uso – Florícola.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

2.3.2 Diagrama de actividades

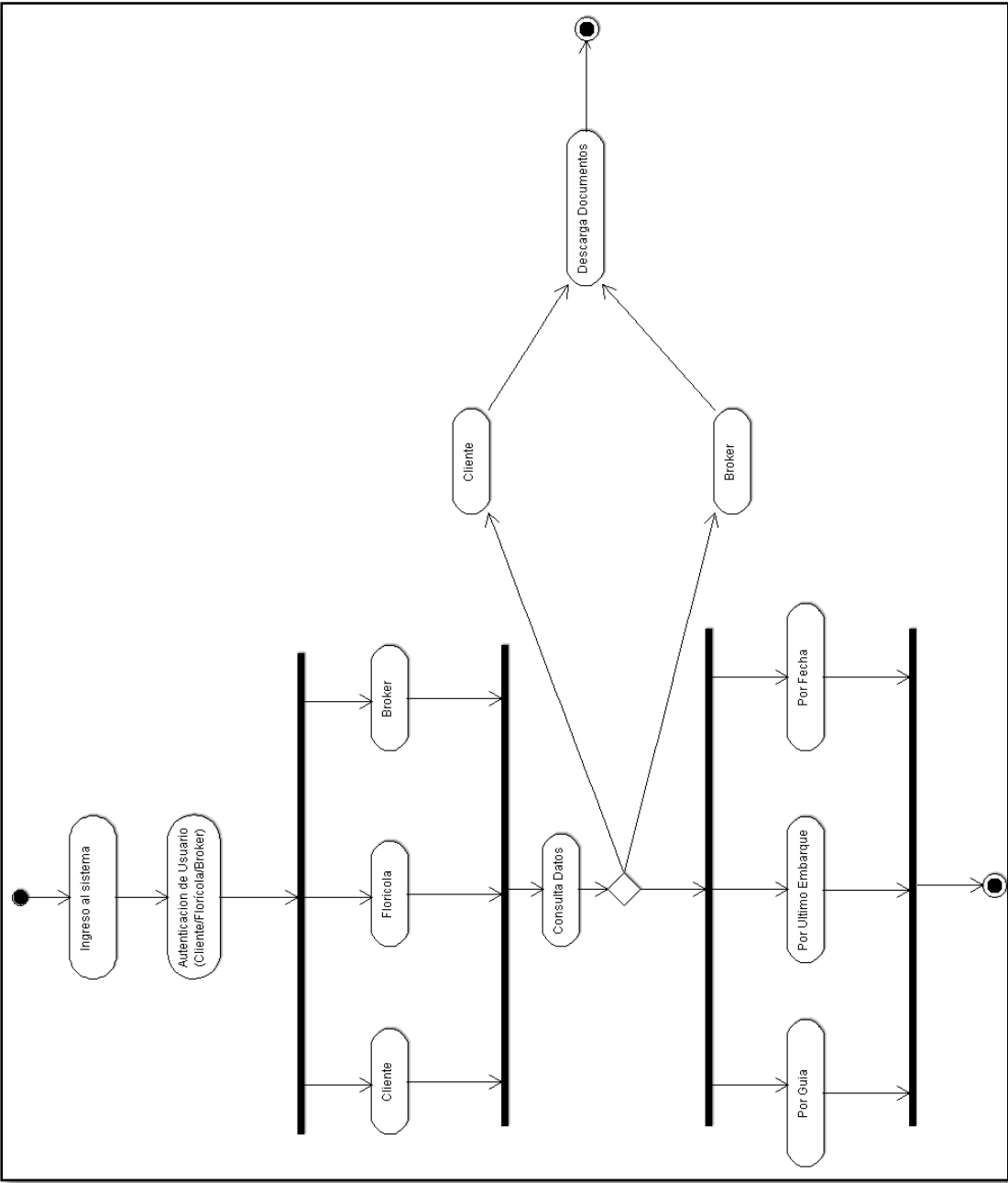


Gráfico 24. Diagrama en Actividades.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

CAPÍTULO III.

DISEÑO

3.1 Diagramas UML

3.1.1 Diagrama de clases

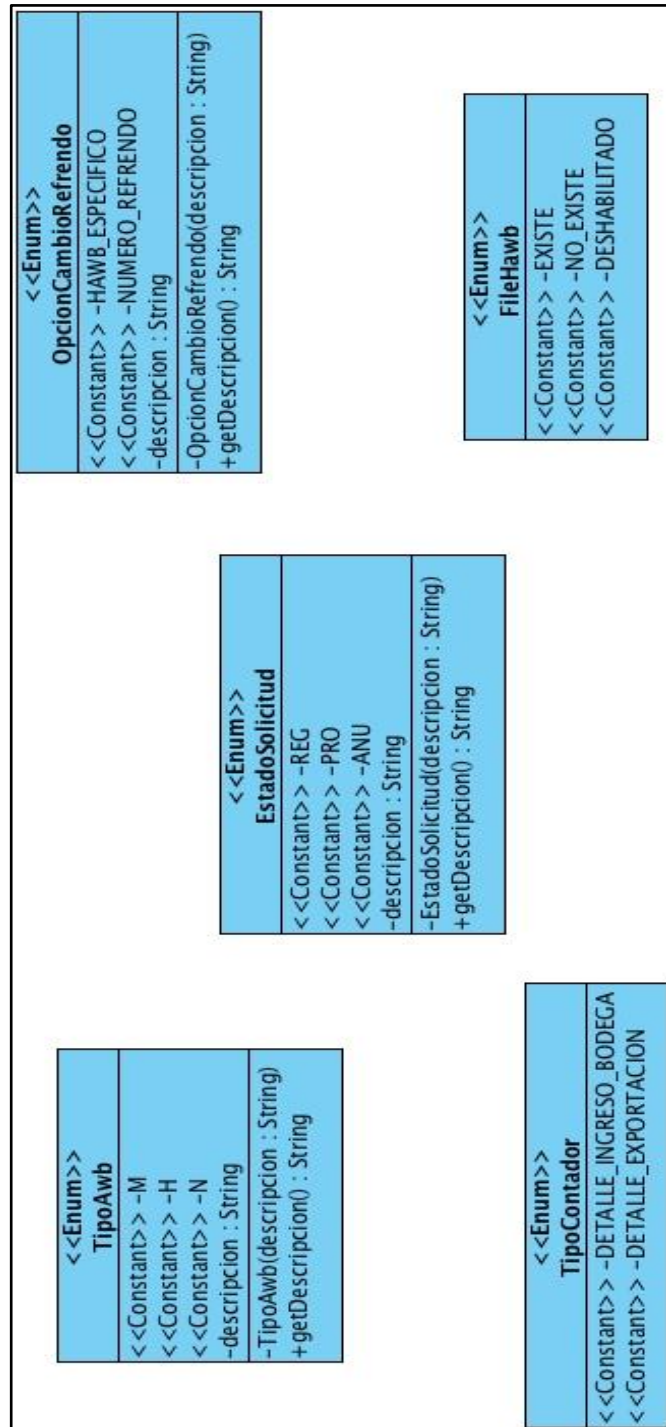


Gráfico 25. a) Diagrama de Constantes; **Sección:** Constantes.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

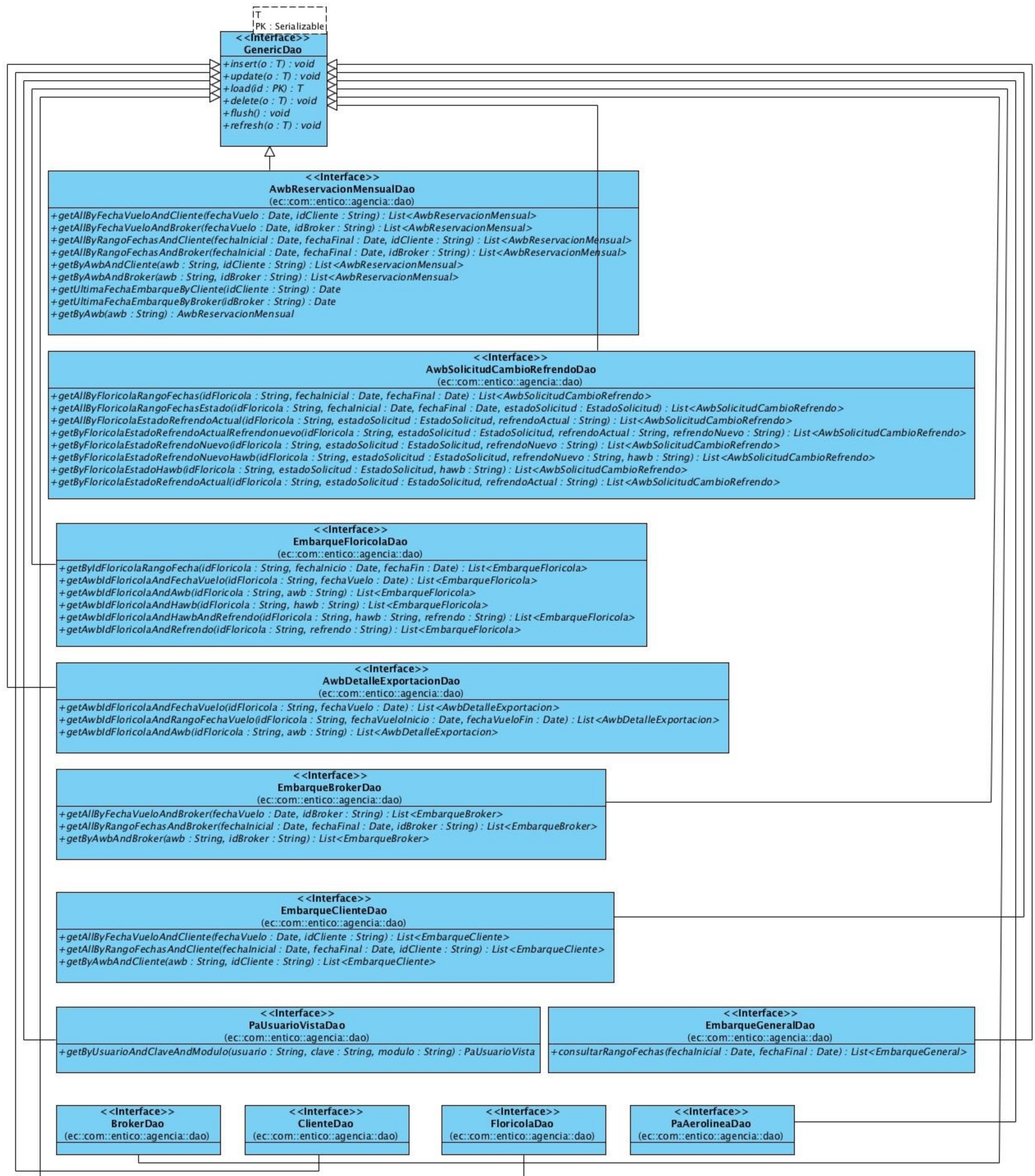


Gráfico 26. b) Diagrama de Constantes; Sección: Dao.
Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cia. Ltda.
Autor: Sánchez Gallardo José.

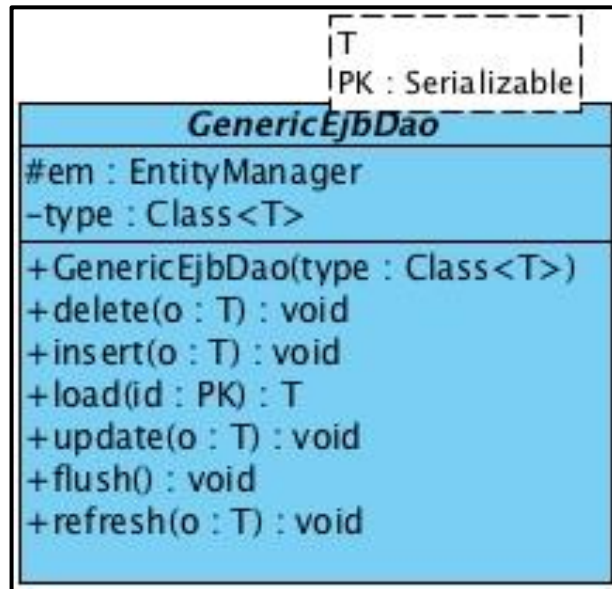


Gráfico 27. c) Diagrama de Constantes; **Sección:** Dao Ejb.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

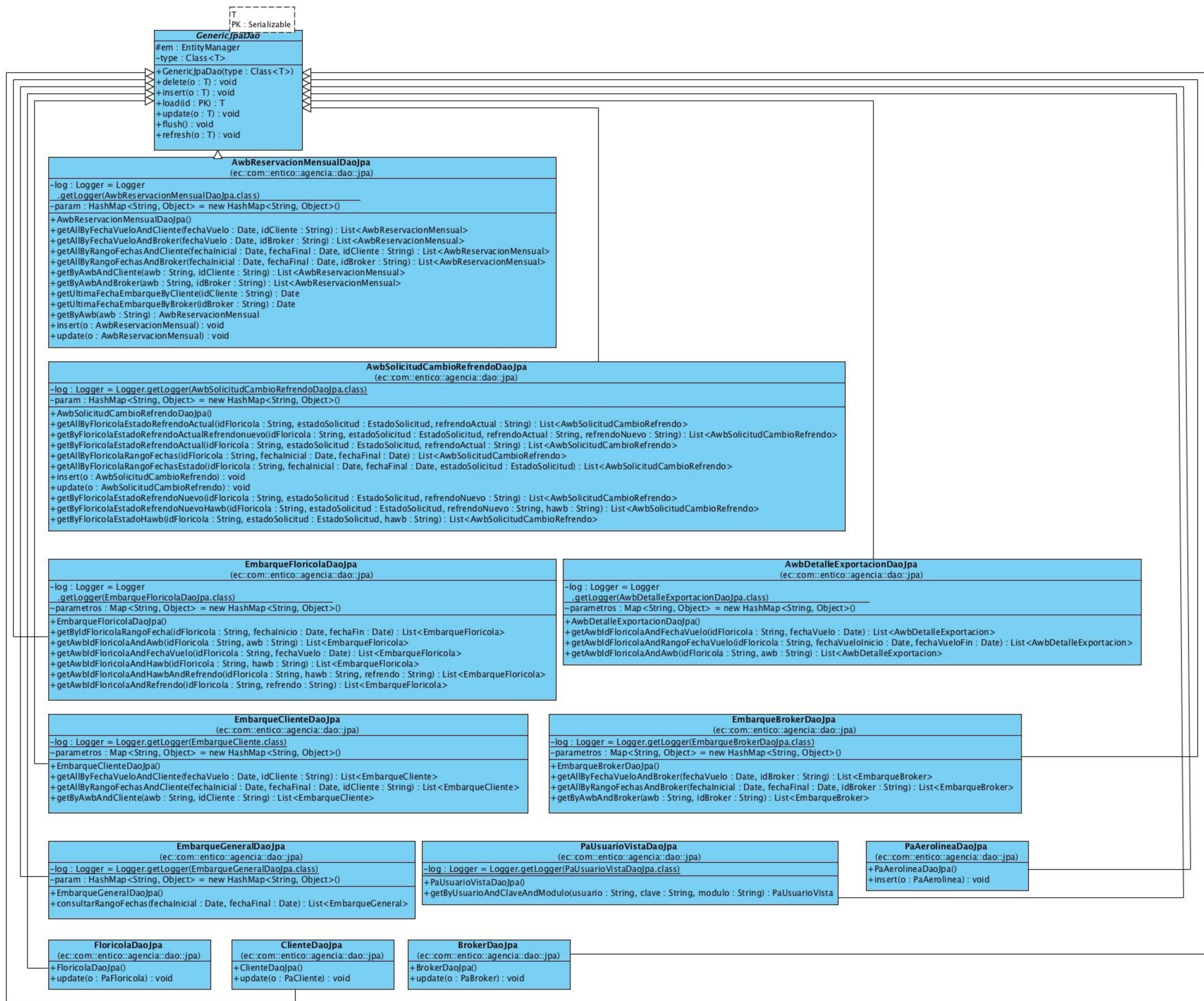


Gráfico 28. d) Diagrama de Constantes; Sección: Dao Jpa.
Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cia. Ltda.
Autor: Sánchez Gallardo José.

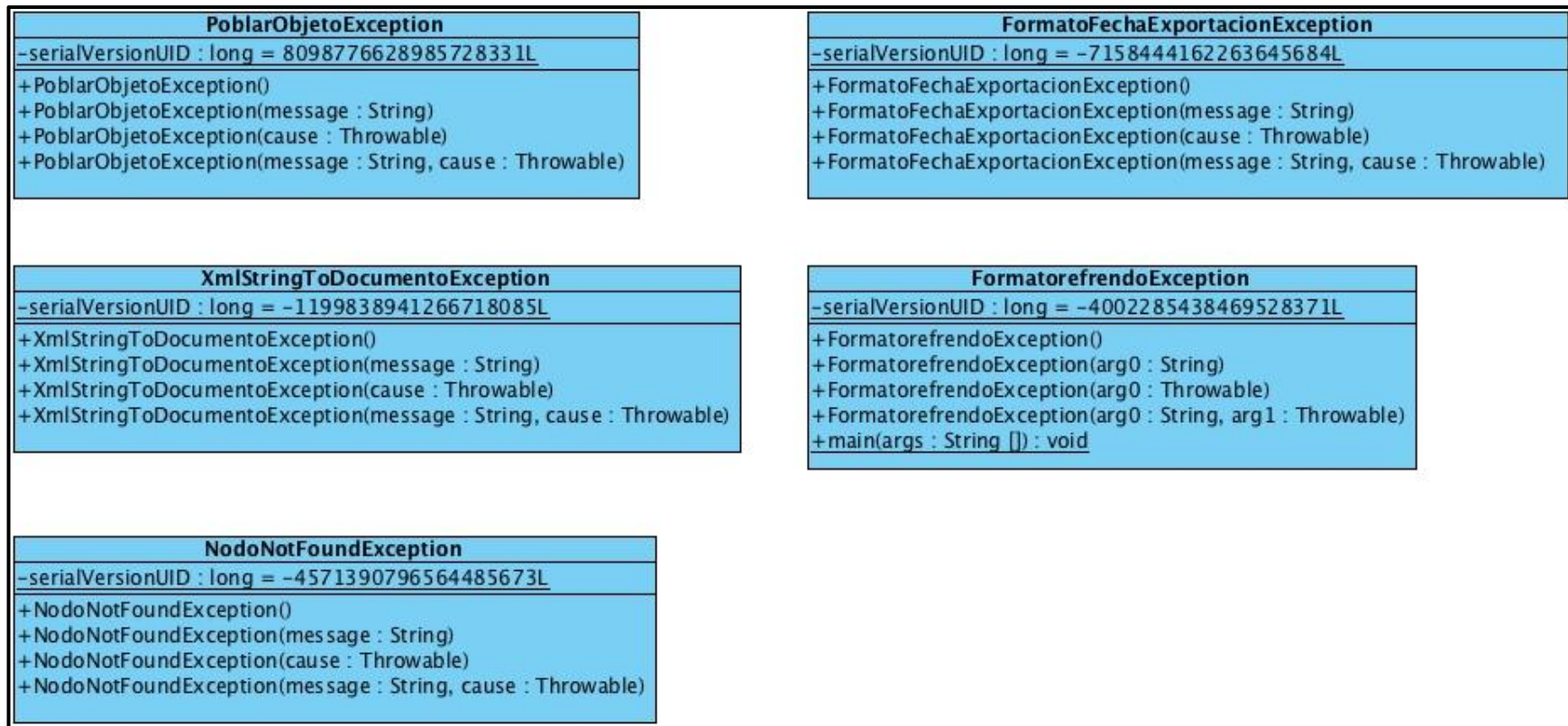


Gráfico 29. e) Diagrama de Constantes; **Sección:** Excepciones.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

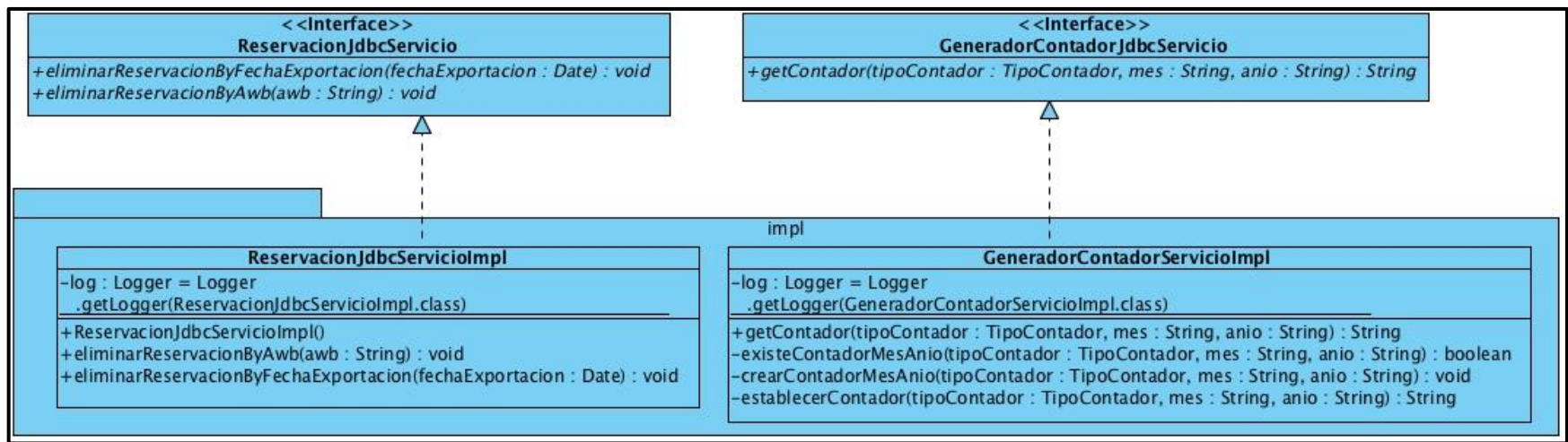


Gráfico 30. f) Diagrama de Constantes; **Sección:** Jdbc.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

EmbarqueFloricola	EmbarqueCliente	EmbarqueBroker	EmbarqueGeneral
<pre> -serialVersionUID : long = -5261233089837750493L -id : String -fechaVuelo : Date -awbOrden : String -awb : String -nombreCliente : String -hawb : String -numeroRefrendo : String -manifiesto : String -cajasreseRvadas : BigDecimal -cajasConfirmadas : BigDecimal -piezasConfirmadas : BigDecimal -peso : BigDecimal -fileHawb : FileHawb = FileHawb.NO_EXISTE -documentoHawb : String = "" +EmbarqueFloricola() +getId() : String +setId(id : String) : void +getFechaVuelo() : Date +setFechaVuelo(fechaVuelo : Date) : void +getAwbOrden() : String +getAwbOrden2() : String +setAwbOrden(awbOrden : String) : void +getAwb() : String +setAwb(awb : String) : void +getNombreCliente() : String +setNombreCliente(nombreCliente : String) : void +getHawb() : String +setHawb(hawb : String) : void +getNumeroRefrendo() : String +setNumeroRefrendo(numeroRefrendo : String) : void +getManifiesto() : String +setManifiesto(manifiesto : String) : void +getCajasreseRvadas() : BigDecimal +setCajasreseRvadas(cajasreseRvadas : BigDecimal) : void +getCajasConfirmadas() : BigDecimal +setCajasConfirmadas(cajasConfirmadas : BigDecimal) : void +getPeso() : BigDecimal +setPeso(peso : BigDecimal) : void +toString() : String +getPiezasConfirmadas() : BigDecimal +setPiezasConfirmadas(piezasConfirmadas : BigDecimal) : void +getFileHawb() : FileHawb +setFileHawb(fileHawb : FileHawb) : void +getDocumentoHawb() : String +setDocumentoHawb(documentoHawb : String) : void +getExisteNumeroManifiesto() : boolean </pre>	<pre> -serialVersionUID : long = 1L -awbOrden : String -awb : String -fecha : Date -nombreCliente : String -cajasReservadas : BigDecimal -cajasOnHand : BigDecimal -piezasReservadas : Long -piezasOnHand : Long -departure : String -arrived : String +EmbarqueCliente() +getAwbOrden() : String +setAwbOrden(awbOrden : String) : void +getNombreCliente() : String +setNombreCliente(nombreCliente : String) : void +getCajasReservadas() : BigDecimal +setCajasReservadas(cajasReservadas : BigDecimal) : void +getCajasOnHand() : BigDecimal +setCajasOnHand(cajasOnHand : BigDecimal) : void +getPiezasReservadas() : Long +setPiezasReservadas(piezasReservadas : Long) : void +getPiezasOnHand() : Long +setPiezasOnHand(piezasOnHand : Long) : void +getDeparture() : String +setDeparture(departure : String) : void +getArrived() : String +setArrived(arrived : String) : void +toString() : String +getAwb() : String +setAwb(awb : String) : void +getFecha() : Date +setFecha(fecha : Date) : void +getAwbOrden2() : String +getAwbCodificada() : String -codif(cad : String) : String </pre>	<pre> -serialVersionUID : long = 1L -awbOrden : String -awb : String -fecha : Date -nombreCliente : String -cajasReservadas : BigDecimal -cajasOnHand : BigDecimal -piezasReservadas : Long -piezasOnHand : Long -departure : String -arrived : String -tipoAwb : TipoAwb +EmbarqueBroker() +getAwbOrden() : String +setAwbOrden(awbOrden : String) : void +getNombreCliente() : String +setNombreCliente(nombreCliente : String) : void +getCajasReservadas() : BigDecimal +setCajasReservadas(cajasReservadas : BigDecimal) : void +getCajasOnHand() : BigDecimal +setCajasOnHand(cajasOnHand : BigDecimal) : void +getPiezasReservadas() : Long +setPiezasReservadas(piezasReservadas : Long) : void +getPiezasOnHand() : Long +setPiezasOnHand(piezasOnHand : Long) : void +getDeparture() : String +setDeparture(departure : String) : void +getArrived() : String +setArrived(arrived : String) : void +toString() : String +getAwb() : String +setAwb(awb : String) : void +getFecha() : Date +setFecha(fecha : Date) : void +getAwbOrden2() : String +getAwbCodificada() : String -codif(cad : String) : String +getTipoAwb() : TipoAwb +setTipoAwb(tipoAwb : TipoAwb) : void </pre>	<pre> -serialVersionUID : long = 1L -awbOrden : String -awb : String -fecha : Date -nombreCliente : String -cajasReservadas : BigDecimal -cajasOnHand : BigDecimal -piezasReservadas : Long -piezasOnHand : Long +EmbarqueGeneral() +getAwbOrden() : String +setAwbOrden(awbOrden : String) : void +getNombreCliente() : String +setNombreCliente(nombreCliente : String) : void +getCajasReservadas() : BigDecimal +setCajasReservadas(cajasReservadas : BigDecimal) : void +getCajasOnHand() : BigDecimal +setCajasOnHand(cajasOnHand : BigDecimal) : void +getPiezasReservadas() : Long +setPiezasReservadas(piezasReservadas : Long) : void +getPiezasOnHand() : Long +setPiezasOnHand(piezasOnHand : Long) : void +toString() : String +getAwb() : String +setAwb(awb : String) : void +getFecha() : Date +setFecha(fecha : Date) : void +getAwbOrden2() : String +getAwbCodificada() : String -codif(cad : String) : String </pre>

Gráfico 31. g) Diagrama de Constantes; **Sección:** Modelo Consulta.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

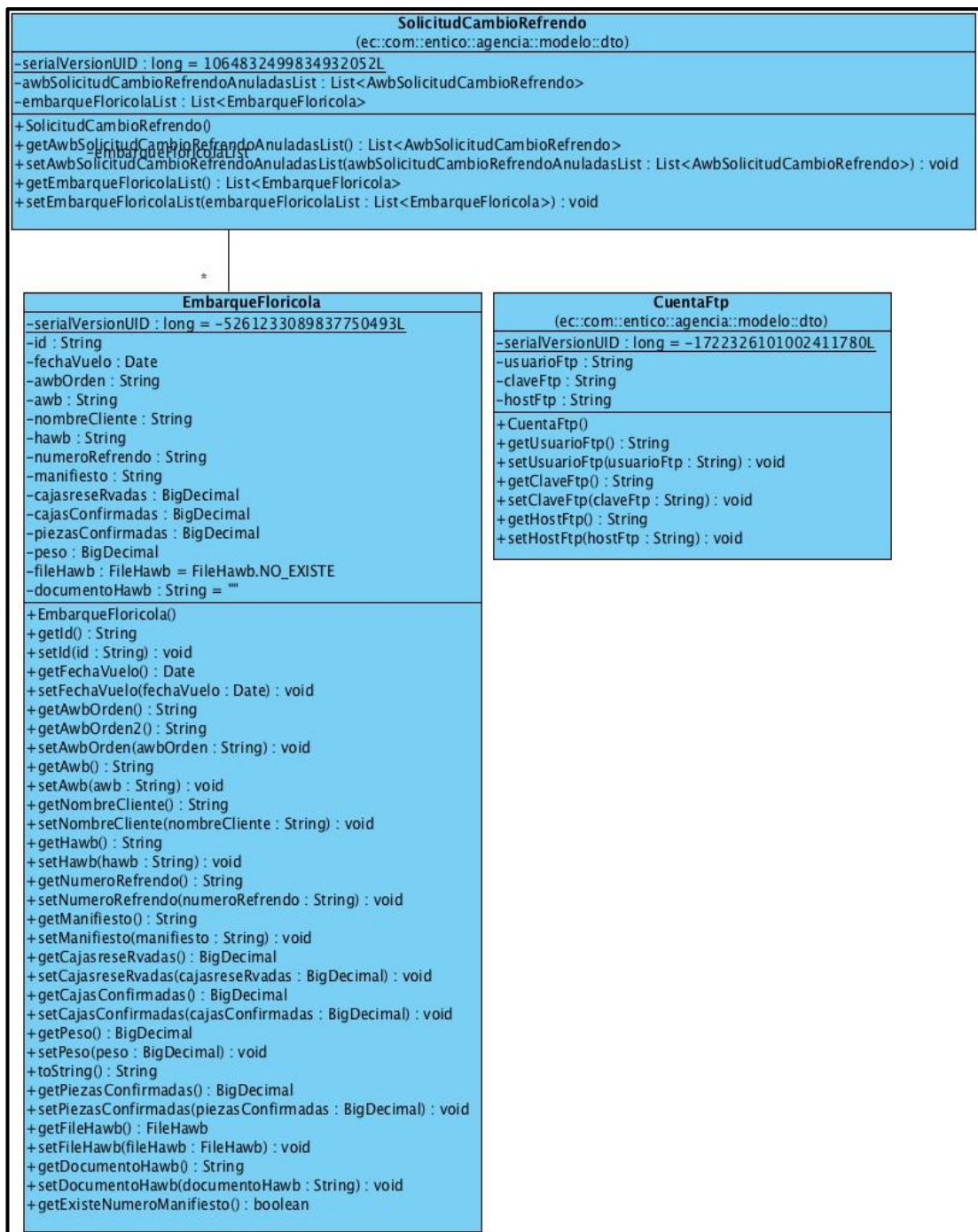


Gráfico 32. h) Diagrama de Constantes; Sección: Modelo Dto.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.



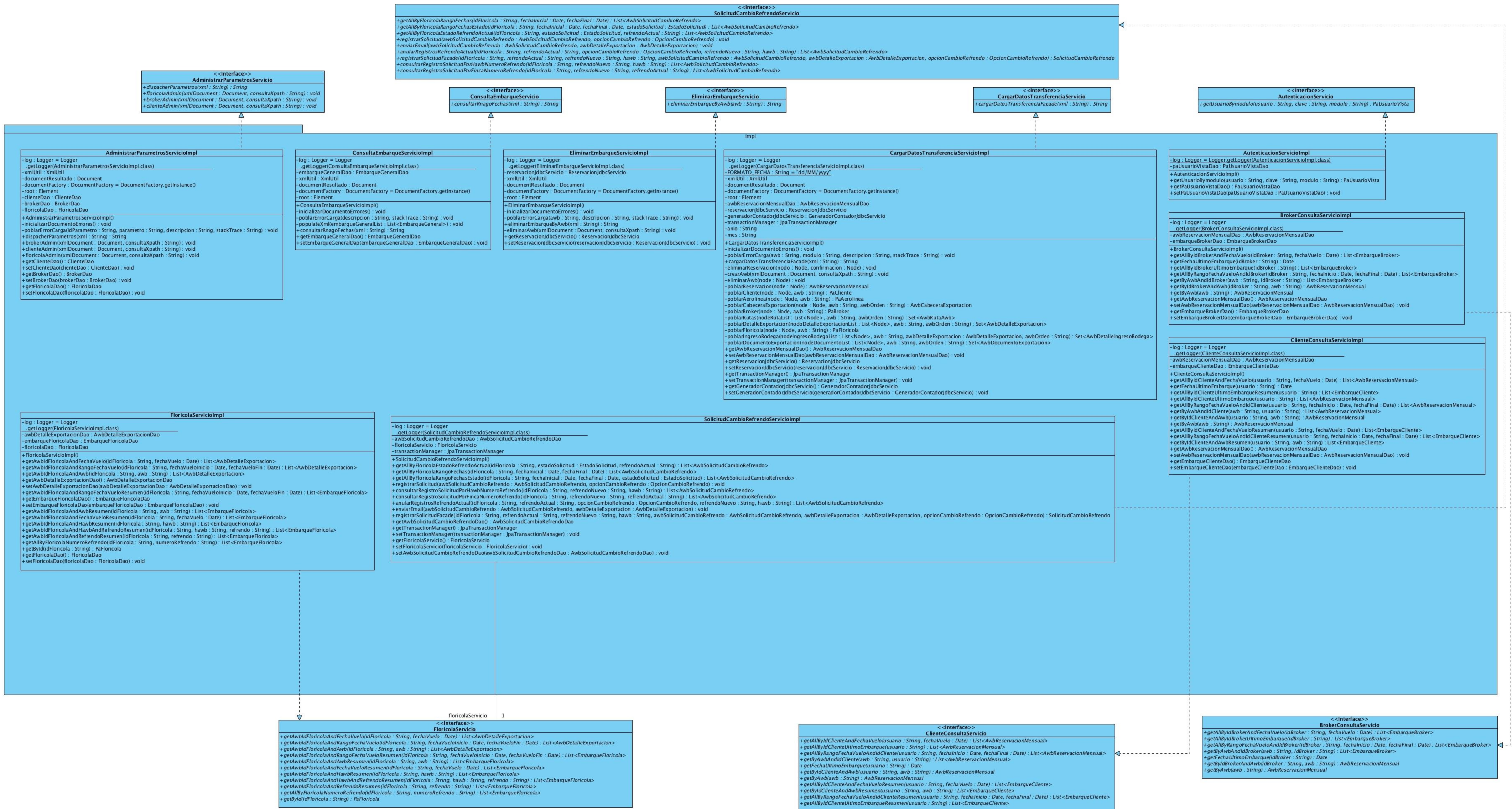


Gráfico 34. Diagrama de Constantes; Sección: Servicios.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.


```

- log : Logger = Logger
- get Logger (FloricolaServicioImpl.class)
- awbDetalleExportacionDao : AwbDetalleExportacionDao
- embarqueFloricolaDao : EmbarqueFloricolaDao
- floricolaDao : FloricolaDao

+ FloricolaServicioImpl()
+ getAwbIdFloricolaAndFechaVuelo(idFloricola : String, fechaVuelo : Date) : List<AwbDetalleExportacion>
+ getAwbIdFloricolaAndRangoDeFechaVuelo(idFloricola : String, fechaVueloInicio : Date, fechaVueloFin : Date) : List<AwbDetalleExportacion>
+ getAwbIdFloricolaAndAwb(idFloricola : String, awb : String) : List<AwbDetalleExportacion>
+ getAwbDetalleExportacionDao() : AwbDetalleExportacionDao
+ setAwbDetalleExportacionDao(awbDetalleExportacionDao : AwbDetalleExportacionDao) : void
+ getAwbIdFloricolaAndRangoDeFechaVueloResumen(idFloricola : String, fechaVueloInicio : Date, fechaVueloFin : Date) : List<EmbarqueFloricola>
+ getEmbarqueFloricolaDao() : EmbarqueFloricolaDao
+ setEmbarqueFloricolaDao(embarqueFloricolaDao : EmbarqueFloricolaDao) : void
+ getAwbIdFloricolaAndAwbResumen(idFloricola : String, awb : String) : List<EmbarqueFloricola>
+ getAwbIdFloricolaAndFechaVueloResumen(idFloricola : String, fechaVuelo : Date) : List<EmbarqueFloricola>
+ getAwbIdFloricolaAndHawbResumen(idFloricola : String, hawb : String) : List<EmbarqueFloricola>
+ getAwbIdFloricolaAndHawbAndRefrendoResumen(idFloricola : String, hawb : String, refrendo : String) : List<EmbarqueFloricola>
+ getAwbIdFloricolaAndRefrendoResumen(idFloricola : String, refrendo : String) : List<EmbarqueFloricola>
+ getAllByFloricolaNumeroRefrendo(idFloricola : String, numeroRefrendo : String) : List<EmbarqueFloricola>
+ getById(idFloricola : String) : PaFloricola
+ getFloricolaDao() : FloricolaDao
+ setFloricolaDao(floricolaDao : FloricolaDao) : void

```

```

-eliminarEmbarqueServicioImpl
- log : Logger = Logger
- getLogger(EliminarEmbarqueServicioImpl.class)
- reservacionJdbcServicio : ReservacionJdbcServicio
- xmlUtil : XmlUtil
- documentResultado : Document
- documentFactory : DocumentFactory = DocumentFactory.getInstance()
- root : Element

+ EliminarEmbarqueServicioImpl()
+ inicializarDocumentoErrores() : void
+ poblarErrorCarga(awb : String, description : String, stackTrace : String) : void
+ eliminarEmbarqueBbyAwb(xml : String) : String
- eliminarAwb(xmlDocumento : Document, consultaXPath : String) : void
+ getReservacionJdbcServicio() : ReservacionJdbcServicio
+ setReservacionJdbcServicio(reservacionJdbcServicio : ReservacionJdbcServicio) : void

```

```

+ ClienteConsultaServicioImpl
- log : Logger = Logger
+ getLogger(ClienteConsultaServicioImpl.class)
- awbReservacionMensualDao : AwbReservacionMensualDao
- embarqueClienteDao : EmbarqueClienteDao
+ ClienteConsultaServicioImpl()
+ getAllyIdClienteAndFechaVuelo(usuario : String, fechaVuelo : Date) : List<AwbReservacionMensual>
+ getFechaUltimoEmbarque(usuario : String) : Date
+ getAllyIdClienteUltimoEmbarqueResumen(usuario : String) : List<EmbarqueCliente>
+ getAllyIdClienteUltimoEmbarque(usuario : String) : List<AwbReservacionMensual>
+ getAllyRangoFechaVueloAndIdCliente(usuario : String, fechaInicio : Date, fechaFinal : Date) : List<AwbReservacionMensual>
+ getFechaVueloAndIdCliente(awb : String, usuario : String) : List<AwbReservacionMensual>
+ getFechaIdClienteAndAwb(usuario : String, awb : String) : AwbReservacionMensual
+ getFechaAwb(awb : String) : AwbReservacionMensual
+ getAllyIdClienteAndFechaVueloResumen(usuario : String, fechaVuelo : Date) : List<EmbarqueCliente>
+ getAllyRangoFechaVueloAndIdClienteResumen(usuario : String, fechaInicio : Date, fechaFinal : Date) : List<EmbarqueCliente>
+ getFechaIdClienteAndAwbResumen(usuario : String, awb : String) : List<EmbarqueCliente>
+ getAwbReservacionMensualDao() : AwbReservacionMensualDao
+ setAwbReservacionMensualDao(awbReservacionMensualDao : AwbReservacionMensualDao) : void
+ setEmbarqueClienteDao() : EmbarqueClienteDao
+ setEmbarqueClienteDao(embarqueClienteDao : EmbarqueClienteDao) : void

```

```

SolitudCambioRefrendoServicioImpl
- log : Logger = Logger
- getLogger(SolitudCambioRefrendoServicioImpl.class)
- awbSolitudCambioRefrendoDao : AwbSolitudCambioRefrendoDao
- floricolaServicio : FloricolaServicio
- transactionManager : JpaTransactionManager
+ SolitudCambioRefrendoServicioImpl()
+ getAllByFloricolaEstadoRefrendoActual(idFloricola : String, estadoSolitud : EstadoSolitud, refrendoActual : String) : List<AwbSolitudCambioRefrendo>
+ getAllByFloricolaRangoFechas(idFloricola : String, fechaInicial : Date, fechaFinal : Date) : List<AwbSolitudCambioRefrendo>
+ getAllByFloricolaRangoFechasEstado(idFloricola : String, fechaInicial : Date, fechaFinal : Date, estadoSolitud : EstadoSolitud) : List<AwbSolitudCambioRefrendo>
+ registrarSolitud(awbSolitudCambioRefrendo : AwbSolitudCambioRefrendo, opcionCambioRefrendo : OpcionCambioRefrendo) : void
+ consultarRegistroSolitudPorHawbNumeroRefrendo(idFloricola : String, refrendoNuevo : String, hawb : String) : List<AwbSolitudCambioRefrendo>
+ consultarRegistroSolitudPorFincaNumeroRefrendo(idFloricola : String, refrendoNuevo : String, refrendoActual : String) : List<AwbSolitudCambioRefrendo>
+ anularRegistrosRefrendoActual(idFloricola : String, refrendoActual : String, opcionCambioRefrendo : OpcionCambioRefrendo, refrendoNuevo : String, hawb : String) : List<AwbSolitudCambioRefrendo>
+ enviarEmail(awbSolitudCambioRefrendo : AwbSolitudCambioRefrendo, awbDetalleExportacion : AwbDetalleExportacion) : void
+ registrarSolitudFacade(idFloricola : String, refrendoActual : String, refrendoNuevo : String, awbSolitudCambioRefrendo : AwbSolitudCambioRefrendo, awbDetalleExportacion : AwbDetalleExportacion, opcionCambioRefrendo : OpcionCambioRefrendo) : SolitudCambioRefrendo
+ getAwbSolitudCambioRefrendoDao() : AwbSolitudCambioRefrendoDao
+ getTransactionManager() : JpaTransactionManager
+ setTransactionManager(transactionManager : JpaTransactionManager) : void
+ getFloricolaServicio() : FloricolaServicio
+ setFloricolaServicio(floricolaServicio : FloricolaServicio) : void
+ setAwbSolitudCambioRefrendoDao(awbSolitudCambioRefrendoDao : AwbSolitudCambioRefrendoDao) : void

```

```

- log : Logger = Logger
- getLogger(CargarDatosTransferenciaServicioImpl.class)
-FORMATO_FECHA : String = "dd/MM/yyyy"
-xmlUtil : XmlUtil
-documentResultado : Document
-documentFactory : DocumentFactory = DocumentFactory.getInstance()
-root : Element
-awbReservacionMensualDao : AwbReservacionMensualDao
-reservacionJdbcServicio : ReservacionJdbcServicio
-generadorContadorJdbcServicio : GeneradorContadorJdbcServicio
-transactionManager : JpaTransactionManager
-anio : String
-mes : String

+CargarDatosTransferenciaServicioImpl()
-inicializarDocumentoErrores() : void
-poblarErrorCarga(awb : String, modulo : String, descripcion : String, stackTrace : String) : void
+ cargarDatosTransferenciaFacade(xml : String) : String
-eliminarReservacion(nodo : Node, confirmation : Node) : void
-crearAwb(xmlDocument : Document, consultaXPath : String) : void
-eliminarAwb(nodo : Node) : void
-poblarReservacion(nodo : Node) : AwbReservacionMensual
-poblarCliente(nodo : Node, awb : String) : PaCliente
-poblarAerolinea(nodo : Node, awb : String) : PaAerolinea
-poblarCabeceraExportacion(nodo : Node, awb : String, awbOrden : String) : AwbCabeceraExportacion
-poblarBroker(nodo : Node, awb : String) : PaBroker
-poblarRutas(nodo RutaList : List<Node>, awb : String, awbOrden : String) : Set<AwbRutaAwb>
-poblarDetalleExportacion(nodoDetalleExportacionList : List<Node>, awb : String, awbOrden : String) : Set<AwbDetalleExportacion>
-poblarFloristica(nodo : Node, awb : String) : PaFloristica
-poblarIngresosBodega(nodoIngresosBodegaList : List<Node>, awb : String, awbDetalleExportacion : AwbDetalleExportacion, awbOrden : String) : Set<AwbDetalleIngresosBodega>
-poblarDocumentoExportacion(nodoDocumentoList : List<Node>, awb : String, awbOrden : String) : Set<AwbDocumentoExportacion>
+getAwbReservacionMensualDao() : AwbReservacionMensualDao
+setAwbReservacionMensualDao(awbReservacionMensualDao : AwbReservacionMensualDao) : void
+getReservacionJdbcServicio() : ReservacionJdbcServicio
+setReservacionJdbcServicio(reservacionJdbcServicio : ReservacionJdbcServicio) : void
+getTransactionManager() : JpaTransactionManager
+setTransactionManager(transactionManager : JpaTransactionManager) : void
+getGeneradorContadorJdbcServicio() : GeneradorContadorJdbcServicio
+setGeneradorContadorJdbcServicio(generadorContadorJdbcServicio : GeneradorContadorJdbcServicio) : void

```

```

AdministrarParametrosServicioImpl

- log : Logger = Logger
- getLogger(AdministrarParametrosServicioImpl.class)
- xmlUtil : XmlUtil
- documentResultado : Document
- documentFactory : DocumentFactory = DocumentFactory.getInstance()
- root : Element
- clienteDao : ClienteDao
- brokerDao : BrokerDao
- floricolaDao : FloricolaDao

+ AdministrarParametrosServicioImpl()
- inicializarDocumentoErrores() : void
- poblarErrorCarga(idParametro : String, parametro : String, descripcion : String, stackTrace : String) : void
+ dispatcherParametros(xml : String) : String
+ brokerAdmin(xmlDocument : Document, consultaXPath : String) : void
+ clienteAdmin(xmlDocument : Document, consultaXPath : String) : void
+ floricolaAdmin(xmlDocument : Document, consultaXPath : String) : void
+ getClienteDao() : ClienteDao
+ setClienteDao(clienteDao : ClienteDao) : void
+ getBrokerDao() : BrokerDao
+ setBrokerDao(brokerDao : BrokerDao) : void
+ getFloricolaDao() : FloricolaDao
+ setFloricolaDao(floricolaDao : FloricolaDao) : void

```

```
-log : Logger = Logger
    _getLogger(ConsultaEmbarqueServicioImpl.class)
-embarqueGeneralDao : EmbarqueGeneralDao
-xmlUtil : XmlUtil
-documentResultado : Document
-documentFactory : DocumentFactory = DocumentFactory.getInstance()
-root : Element

+ConsultaEmbarqueServicioImpl()
-initializaDocumentoErrores() : void
-poblarErrorCarga(description : String, stackTrace : String) : void
-populateXml(embarqueGeneralList : List<EmbarqueGeneral>) : void
+consultaRnagoFechas(xml : String) : String
+getEmbarqueGeneralDao() : EmbarqueGeneralDao
+setEmbarqueGeneralDao(embarqueGeneralDao : EmbarqueGeneralDao) : void
```

```

BrokerConsultaServicioImpl
- log : Logger = Logger
  _getLogger(BrokerConsultaServicioImpl.class)
- awbReservacionMensualDao : AwbReservacionMensualDao
  embarqueBrokerDao : EmbarqueBrokerDao
+ BrokerConsultaServicioImpl()
+ getAlByIdBrokerAndFechaVuelo(idBroker : String, fechaVuelo : Date) : List<EmbarqueBroker>
+ getFechaUltimoEmbarque(idBroker : String) : Date
+ getAlByIdBrokerUltimoEmbarque(idBroker : String) : List<EmbarqueBroker>
+ getAlByRangOfFechaVueloAndIdBroker(idBroker : String, fechaInicio : Date, fechaFinal : Date) : List<EmbarqueBroker>
+ getByAwbAndIdBroker(awb : String, idBroker : String) : List<EmbarqueBroker>
+ getByIdBrokerAndAwb(idBroker : String, awb : String) : AwbReservacionMensual
+ getByAwb(awb : String) : AwbReservacionMensual
+ getAwbReservacionMensualDao() : AwbReservacionMensualDao
+ setAwbReservacionMensualDao(awbReservacionMensualDao : AwbReservacionMensualDao) : void
+ getEmbarqueBrokerDao() : EmbarqueBrokerDao
+ setEmbarqueBrokerDao(embarqueBrokerDao : EmbarqueBrokerDao) : void

```

Gráfico: 35. m) Diagrama de Constantes; Sección: Servicios – Implementación.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

<<Interface>> BrokerConsultaServicio
+getAllByIdBrokerAndFechaVuelo(idBroker : String, fechaVuelo : Date) : List<EmbarqueBroker> +getAllByIdBrokerUltimoEmbarque(idBroker : String) : List<EmbarqueBroker> +getAllByRangoFechaVueloAndIdBroker(idBroker : String, fechaInicio : Date, fechaFinal : Date) : List<EmbarqueBroker> +getByAwbAndIdBroker(awb : String, idBroker : String) : List<EmbarqueBroker> +getFechaUltimoEmbarque(idBroker : String) : Date +getByIdBrokerAndAwb(idBroker : String, awb : String) : AwbReservacionMensual +getByAwb(awb : String) : AwbReservacionMensual

<<Interface>> AdministrarParametrosServicio
+dispatcherParametros(xml : String) : String +floricolaAdmin(xmlDocument : Document, consultaXpath : String) : void +brokerAdmin(xmlDocument : Document, consultaXpath : String) : void +clienteAdmin(xmlDocument : Document, consultaXpath : String) : void

<<Interface>> FloricolaServicio
+getAwbIdFloricolaAndFechaVuelo(idFloricola : String, fechaVuelo : Date) : List<AwbDetalleExportacion> +getAwbIdFloricolaAndRangoFechaVuelo(idFloricola : String, fechaVueloInicio : Date, fechaVueloFin : Date) : List<AwbDetalleExportacion> +getAwbIdFloricolaAndAwb(idFloricola : String, awb : String) : List<AwbDetalleExportacion> +getAwbIdFloricolaAndRangoFechaVueloResumen(idFloricola : String, fechaVueloInicio : Date, fechaVueloFin : Date) : List<EmbarqueFloricola> +getAwbIdFloricolaAndAwbResumen(idFloricola : String, awb : String) : List<EmbarqueFloricola> +getAwbIdFloricolaAndFechaVueloResumen(idFloricola : String, fechaVuelo : Date) : List<EmbarqueFloricola> +getAwbIdFloricolaAndHawbResumen(idFloricola : String, hawb : String) : List<EmbarqueFloricola> +getAwbIdFloricolaAndHawbAndRefrendoResumen(idFloricola : String, hawb : String, refrendo : String) : List<EmbarqueFloricola> +getAwbIdFloricolaAndRefrendoResumen(idFloricola : String, refrendo : String) : List<EmbarqueFloricola> +getAllByFloricolaNumeroRefrendo(idFloricola : String, numeroRefrendo : String) : List<EmbarqueFloricola> +getById(idFloricola : String) : PaFloricola

<<Interface>> SolicitudCambioRefrendoServicio
+getAllByFloricolaRangoFechas(idFloricola : String, fechaInicial : Date, fechaFinal : Date) : List<AwbSolicitudCambioRefrendo> +getAllByFloricolaRangoFechasEstado(idFloricola : String, fechaInicial : Date, fechaFinal : Date, estadoSolicitud : EstadoSolicitud) : List<AwbSolicitudCambioRefrendo> +getAllByFloricolaEstadoRefrendoActual(idFloricola : String, estadoSolicitud : EstadoSolicitud, refrendoActual : String) : List<AwbSolicitudCambioRefrendo> +registrarSolicitud(awbSolicitudCambioRefrendo : AwbSolicitudCambioRefrendo, opcionCambioRefrendo : OpcionCambioRefrendo) : void +enviarEmail(awbSolicitudCambioRefrendo : AwbSolicitudCambioRefrendo, awbDetalleExportacion : AwbDetalleExportacion) : void +anularRegistrosRefrendoActual(idFloricola : String, refrendoActual : String, opcionCambioRefrendo : OpcionCambioRefrendo, refrendoNuevo : String, hawb : String) : List<AwbSolicitudCambioRefrendo> +registrarSolicitudFacade(idFloricola : String, refrendoActual : String, refrendoNuevo : String, hawb : String, awbSolicitudCambioRefrendo : AwbSolicitudCambioRefrendo, awbDetalleExportacion : AwbDetalleExportacion, opcionCambioRefrendo : OpcionCambioRefrendo) : SolicitudCambioRefrendo +consultarRegistroSolicitudPorHawbNumeroRefrendo(idFloricola : String, refrendoNuevo : String, hawb : String) : List<AwbSolicitudCambioRefrendo> +consultarRegistroSolicitudPorFincaNumeroRefrendo(idFloricola : String, refrendoNuevo : String, refrendoActual : String) : List<AwbSolicitudCambioRefrendo>

<<Interface>> ClienteConsultaServicio
+getAllByIdClienteAndFechaVuelo(usuario : String, fechaVuelo : Date) : List<AwbReservacionMensual> +getAllByIdClienteUltimoEmbarque(usuario : String) : List<AwbReservacionMensual> +getAllByRangoFechaVueloAndIdCliente(usuario : String, fechaInicio : Date, fechaFinal : Date) : List<AwbReservacionMensual> +getByAwbAndIdCliente(awb : String, usuario : String) : List<AwbReservacionMensual> +getFechaUltimoEmbarque(usuario : String) : Date +getByIdClienteAndAwb(usuario : String, awb : String) : AwbReservacionMensual +getByAwb(awb : String) : AwbReservacionMensual +getAllByIdClienteAndFechaVueloResumen(usuario : String, fechaVuelo : Date) : List<EmbarqueCliente> +getByIdClienteAndAwbResumen(usuario : String, awb : String) : List<EmbarqueCliente> +getAllByRangoFechaVueloAndIdClienteResumen(usuario : String, fechaInicio : Date, fechaFinal : Date) : List<EmbarqueCliente> +getAllByIdClienteUltimoEmbarqueResumen(usuario : String) : List<EmbarqueCliente>

<<Interface>> ConsultaEmbarqueServicio
+consultarRnagoFechas(xml : String) : String

<<Interface>> EliminarEmbarqueServicio
+eliminarEmbarqueByAwb(awb : String) : String

<<Interface>> AutenticacionServicio
+getUsuarioBymodulo(usuario : String, clave : String, modulo : String) : PaUsuarioVista

<<Interface>> CargarDatosTransferenciaServicio
+cargarDatosTransferenciaFacade(xml : String) : String

Gráfico 36. n) Diagrama de Constantes; Sección: Servicios – Interfaces.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.



Gráfico 37. o) Diagrama de Constantes; **Sección:** Utilitarios.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

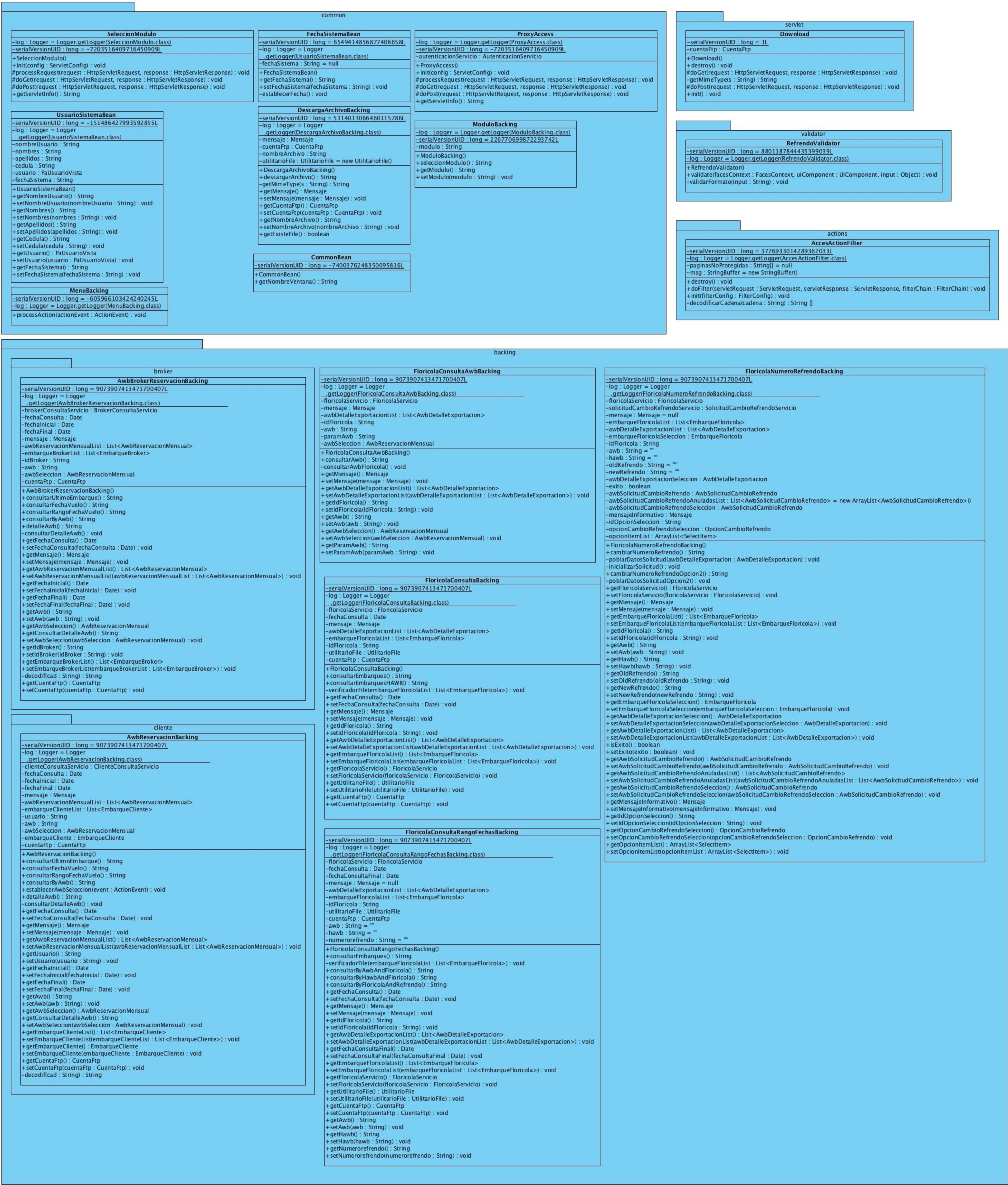


Gráfico 38. p) Diagrama de Constantes; Sección: Web.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cia. Ltda.

Autor: Sánchez Gallardo José.

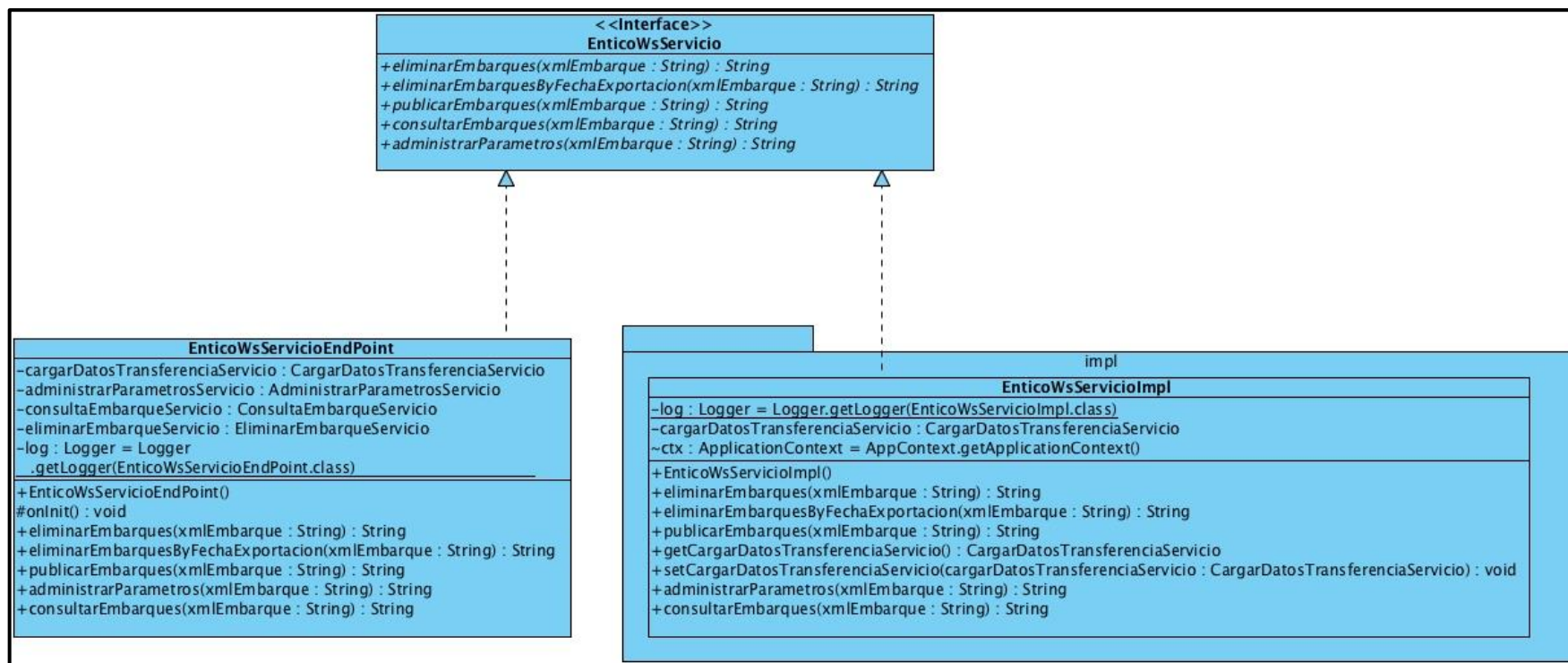


Gráfico 39. Diagrama de Constantes; **Sección:** Web Service.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

3.1.2 Diagrama de objetos

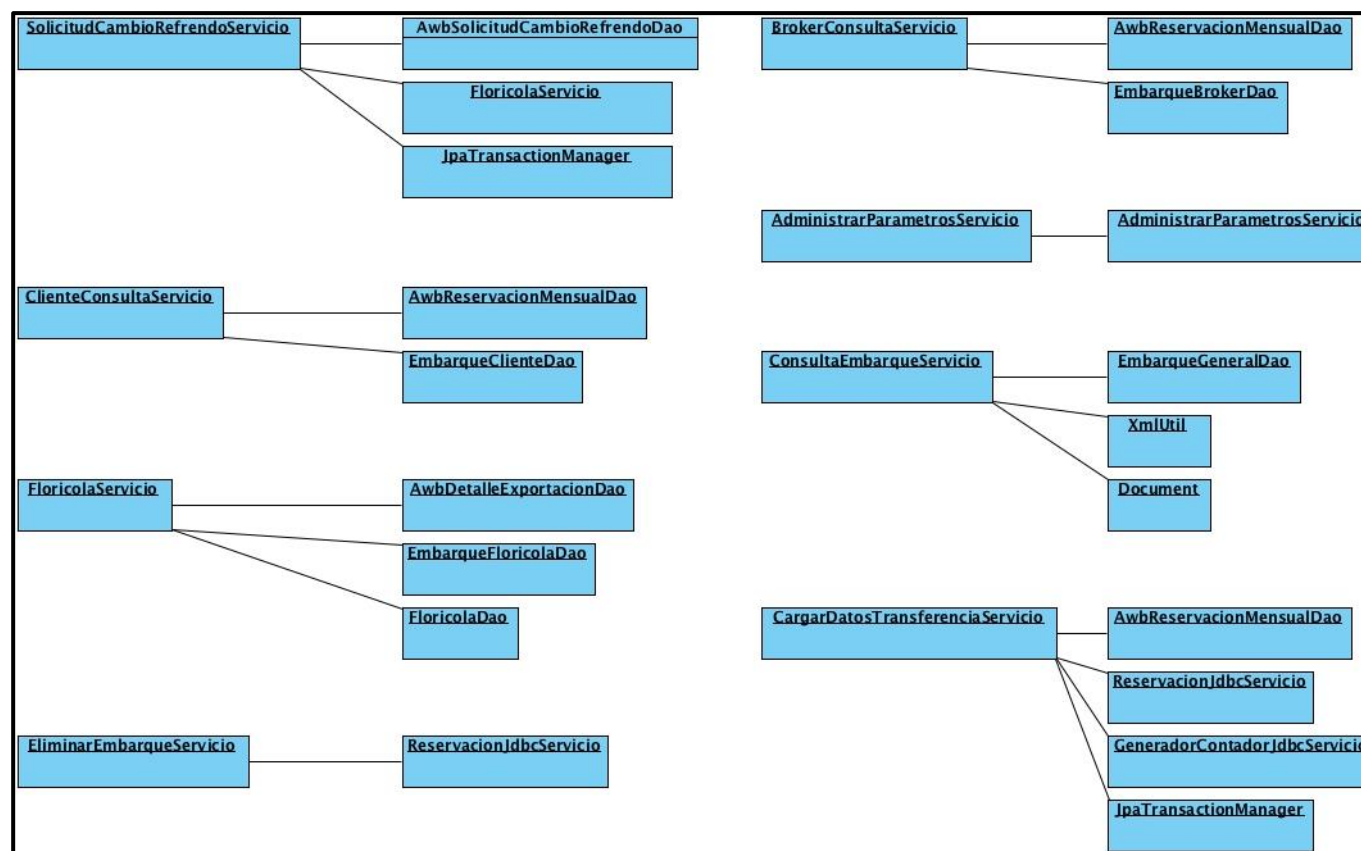


Gráfico 40. Diagrama de Objetos; **Sección:** Refrendos.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

3.1.3 Diagrama de paquetes

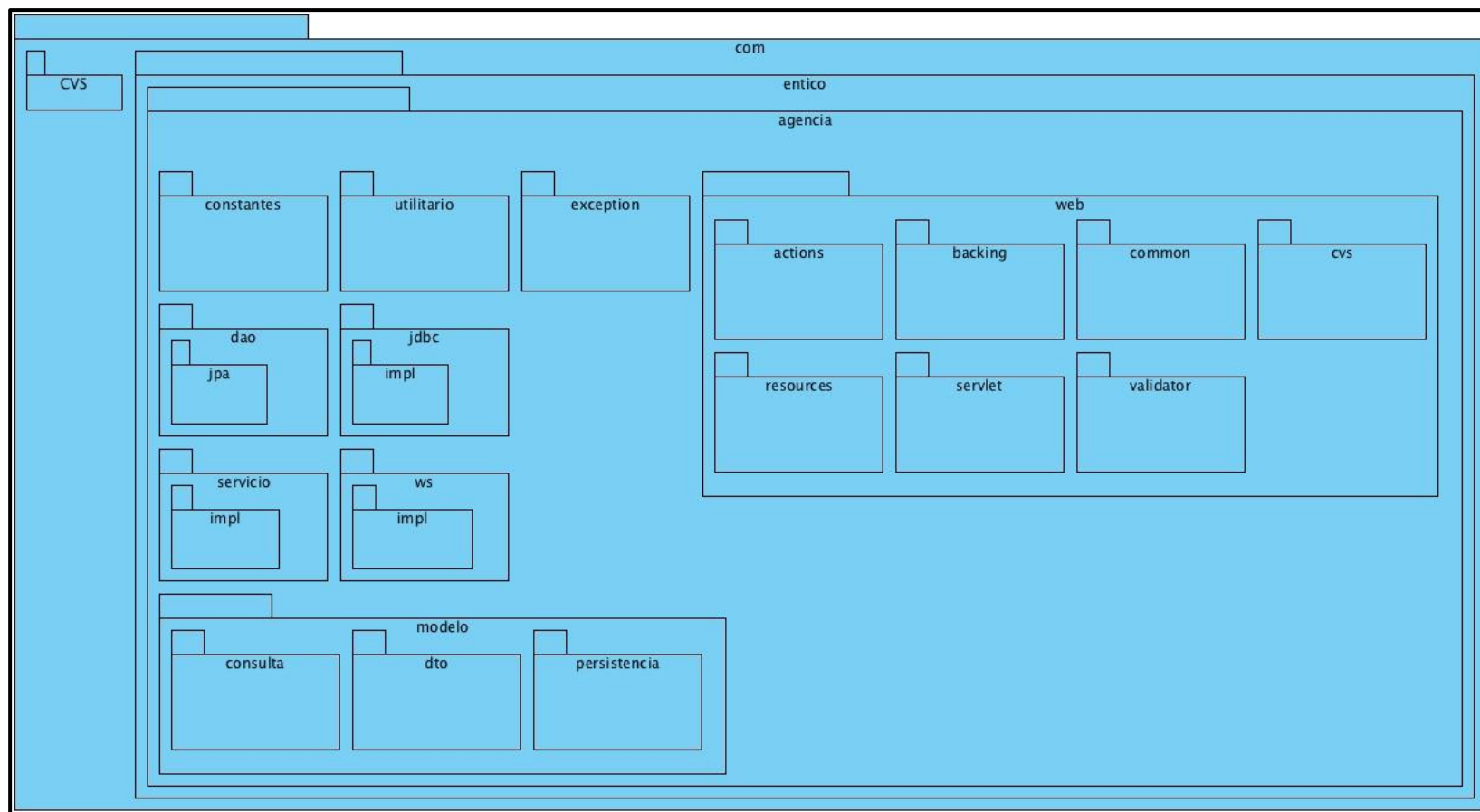


Gráfico 41. Diagrama de Paquetes.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

3.1.4 Diagrama de componentes

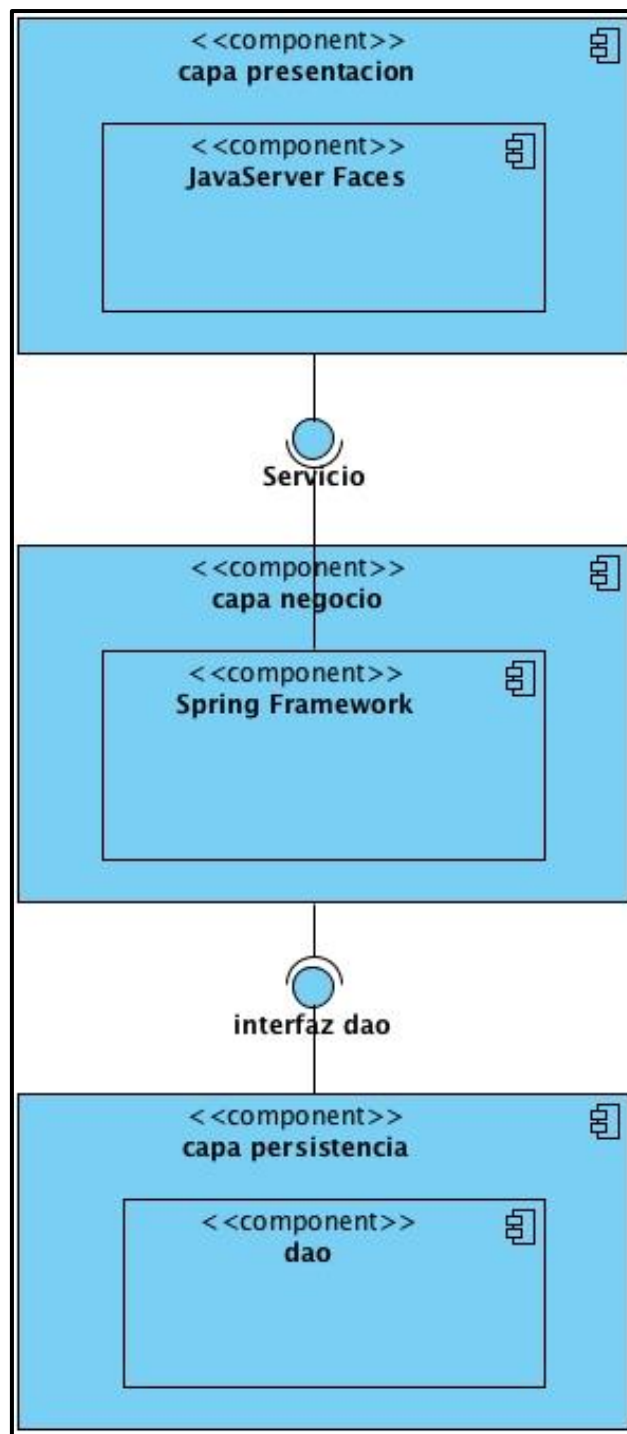


Gráfico 42: Diagrama Componentes.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

3.1.5 Diagrama de secuencia

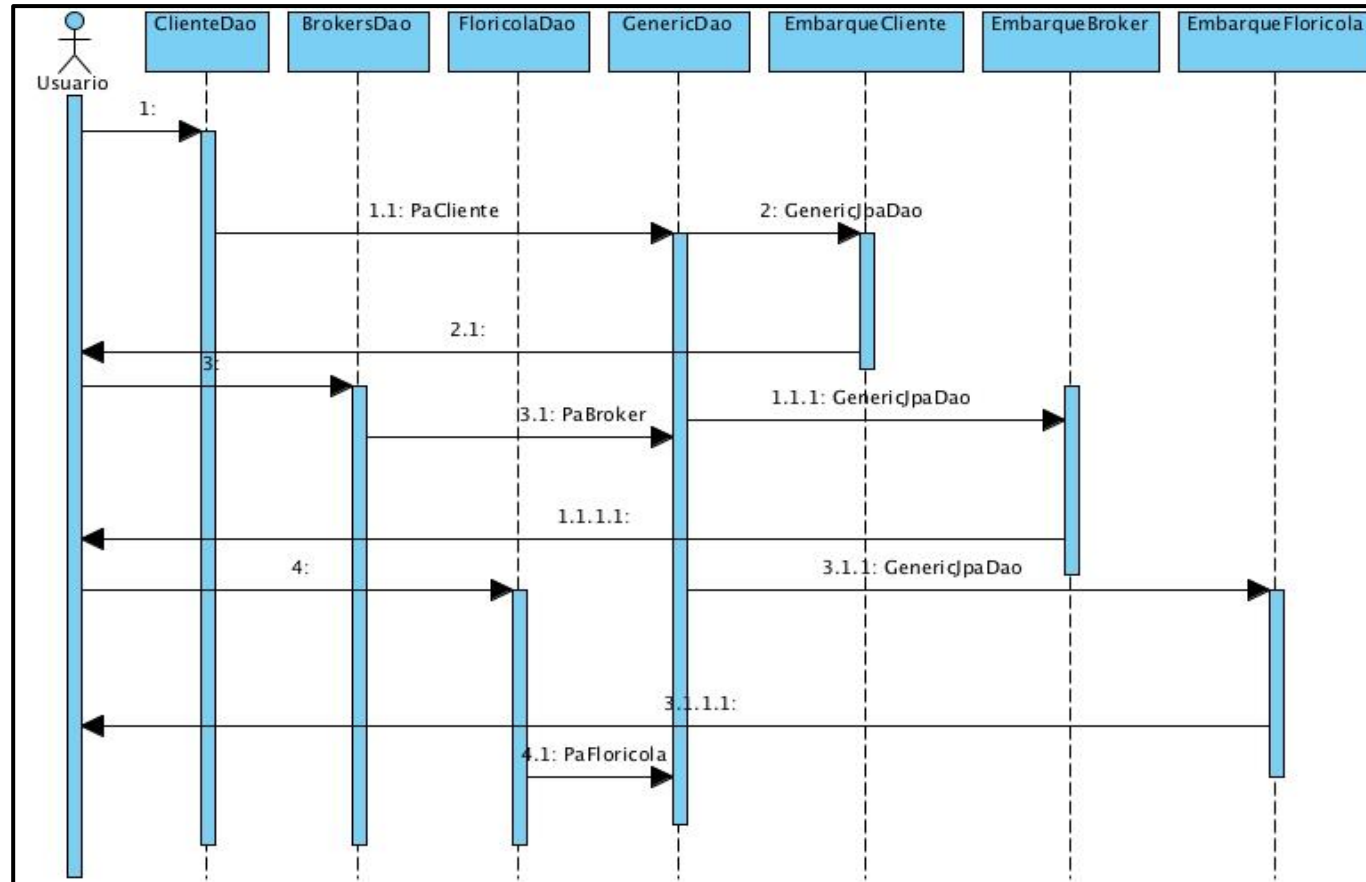


Gráfico 43 a) Diagrama de Secuencia; **Sección:** Administración de Parámetros.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

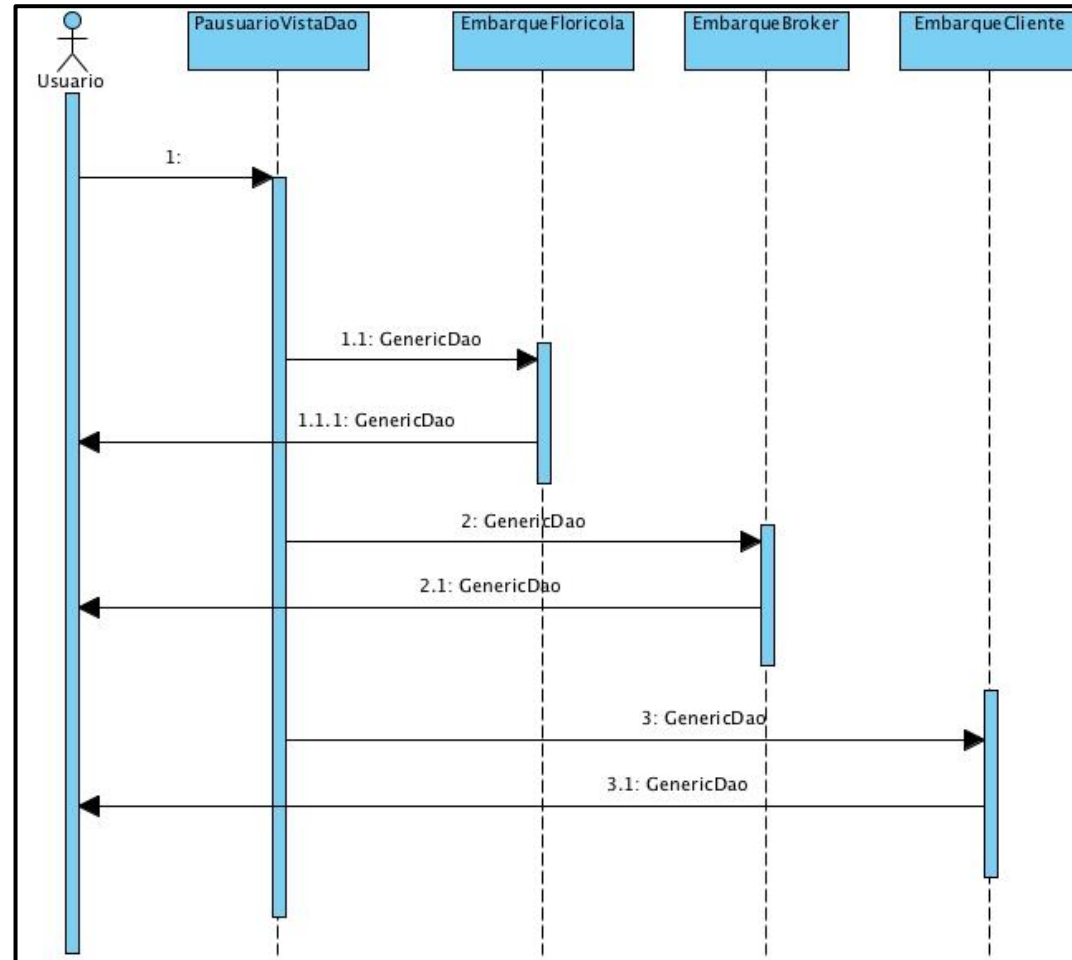


Gráfico 44 b) Diagrama de Secuencia; **Sección:** Autenticación.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

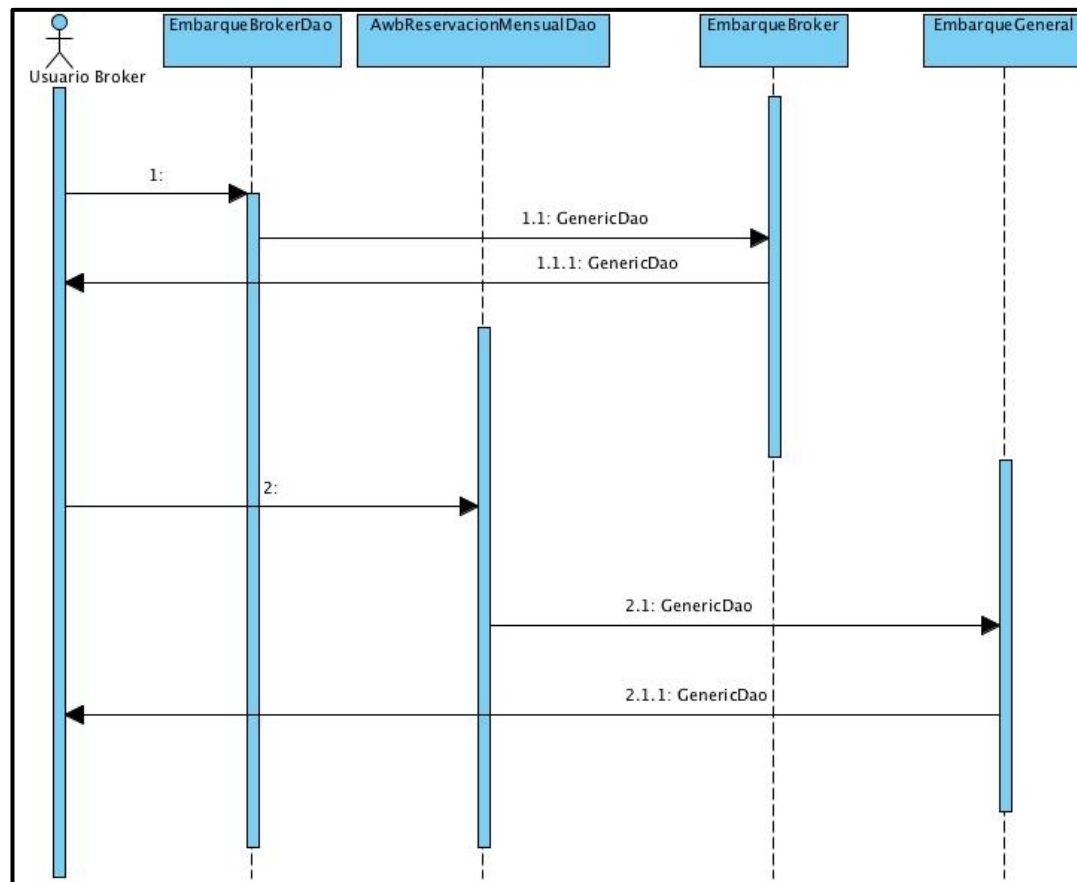


Gráfico 45 c) Diagrama de Secuencia; **Sección:** Broker – Consulta.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

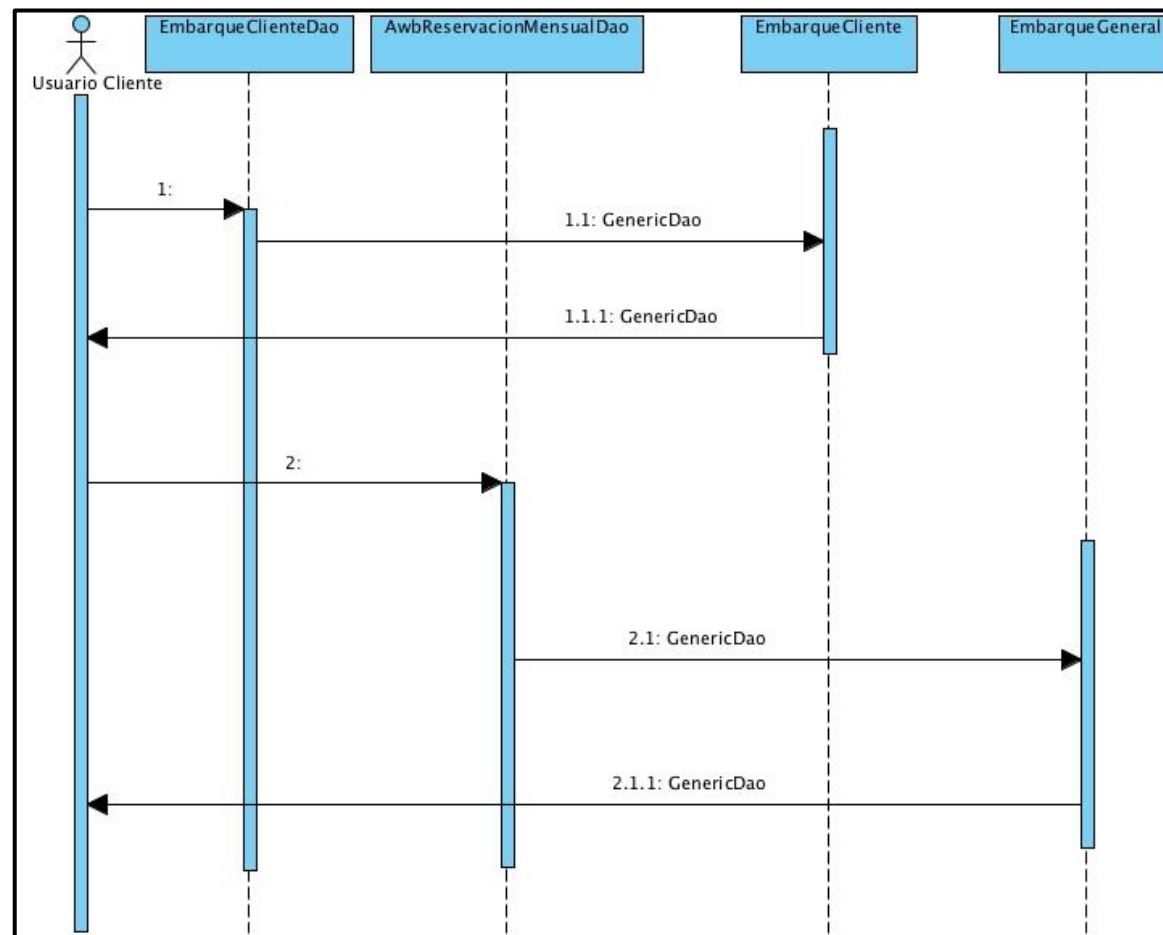


Gráfico 46 d) Diagrama de Secuencia; **Sección:** Cliente – Consulta.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

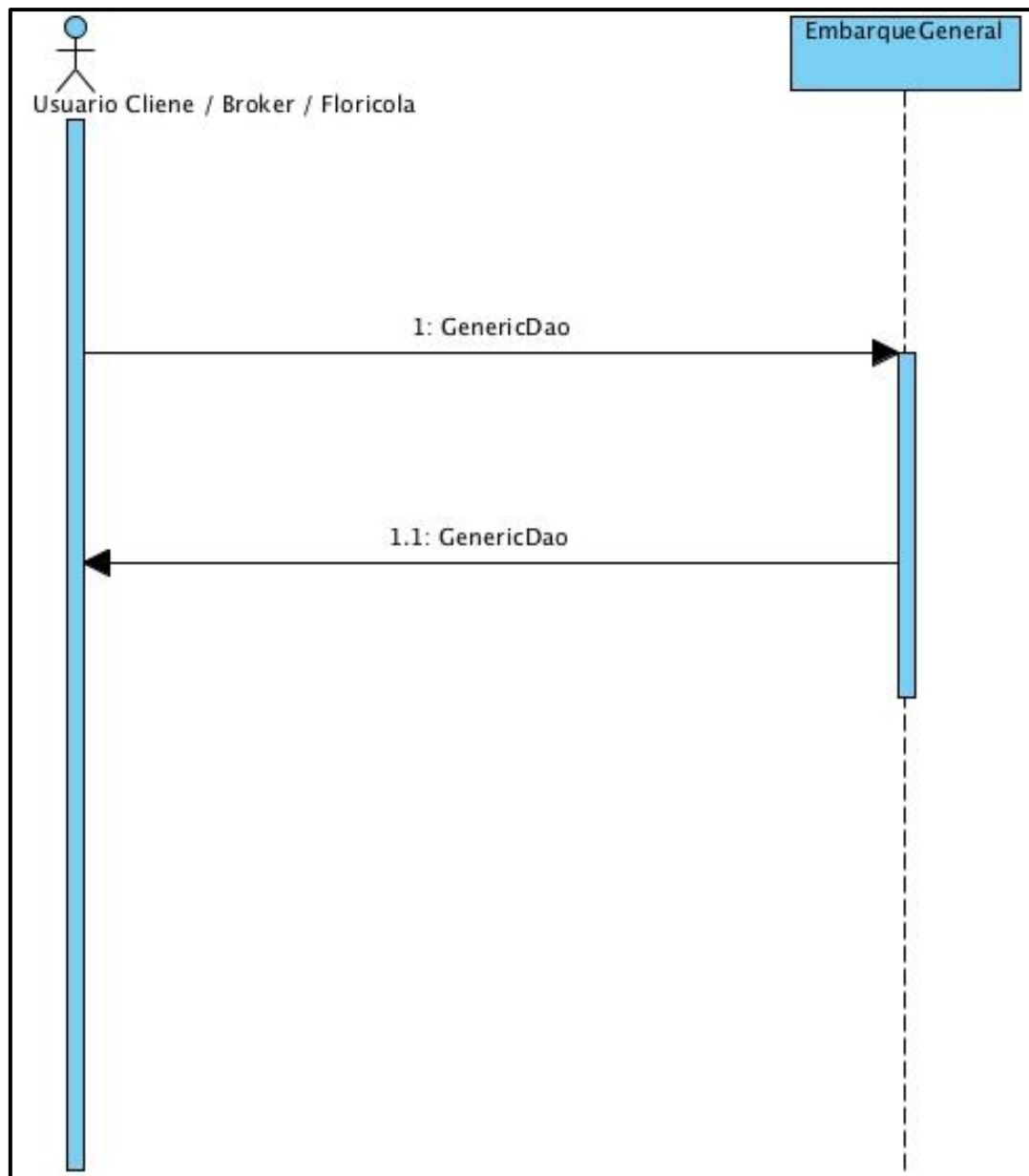


Gráfico 47. e) Diagrama de Secuencia; **Sección:** Consulta – Embarque.
Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.
Autor: Sánchez Gallardo José

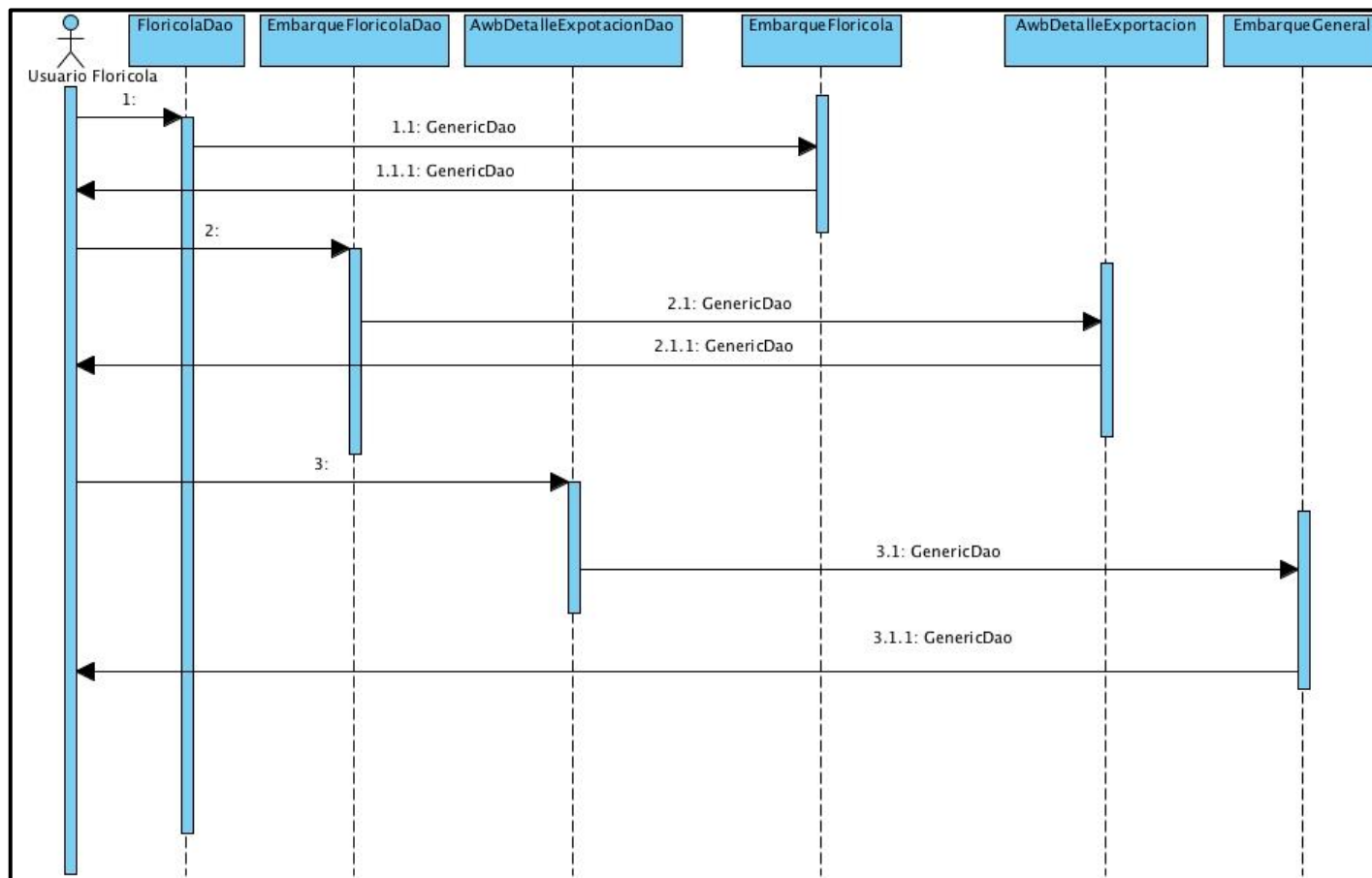


Gráfico 48. f) Diagrama de Secuencia; **Sección:** Florícola – Consulta.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José

3.2 Diseño de bases de datos

3.2.1 Modelo físico

La base de datos diseñada, cuenta con 12 tablas las cuales se detallan a continuación:

a. Tablas del Negocio:

I. Estas tablas están relacionadas entre sí, para mantener la integridad referencial de las mismas:

- awb_reservacionmensual_tbl: contiene la primera información de la reserva de los datos;
- awb_cabeceraexportacion_tbl: contiene la información de las guías con los datos principales de los vuelos;
- awb_detalleexportacion_tbl: contiene la información de las florícolas coordinadas a un determinado cliente;
- awb_detalleingresobodega_tbl: contiene la información de bodega de cada florícola;
- awb_rutaawb_tbl: contiene el desglose con la ruta de vuelo de la guía; y,
- awb_documentoexportacion_tbl: contiene la información con los nombres de los archivos por guía que se suben a la web y que están disponibles para que usuario los pueda o no descargar.

b. Tablas de Parámetros:

II. Estas tablas tienen información que sirve para parametrizar los datos que se encuentran en las tablas del negocio, estas tablas no dependen de otras:

- pa_aerolinea_tbl: contiene la información básica de la aerolínea;
- pa_cliente_tbl: contiene la información básica del bróker;
- pa_broker_tbl: contiene la información básica del bróker;
- pa_floricola_tbl: contiene la información básica de la florícola;

- pa-tipocliente_tbl: contiene la información para especificar si un cliente es bróker u otro tipo de cliente según el negocio de la agencia; y,
- pa_contador_tbl: contiene todos los contadores del sistema, que son las claves principales de cada tabla, que al insertarse, buscan su nueva clave en esta tabla.

A continuación se presentan los diagramas de la base de datos, el modelo físico que está en funcionamiento:

a. Tablas del negocio:

En estas tablas se almacena la información que va a ser transferida desde el sistema cliente – servidor de la agencia:

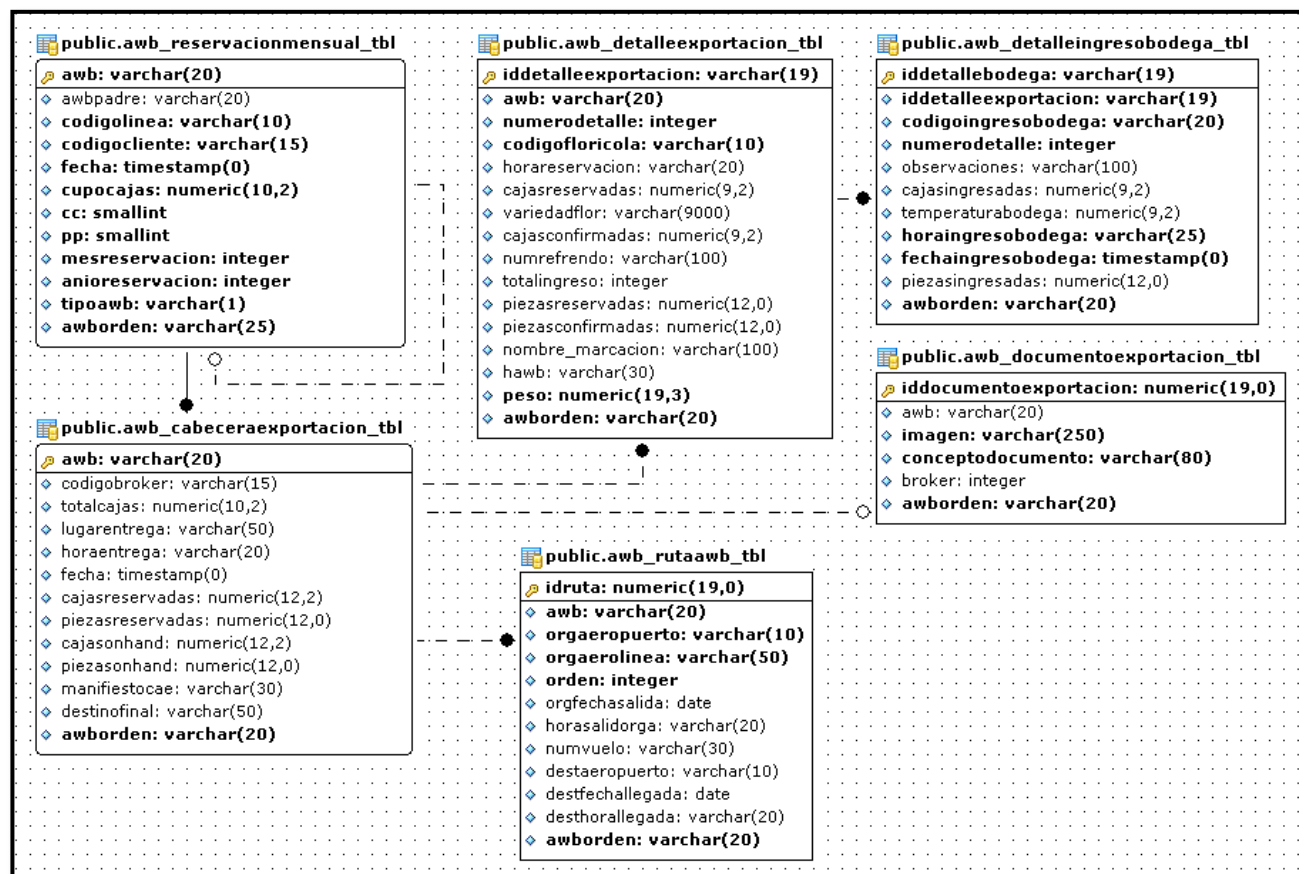


Gráfico 49 literal a) Tablas de Negocio.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

b. Tablas de parámetros:

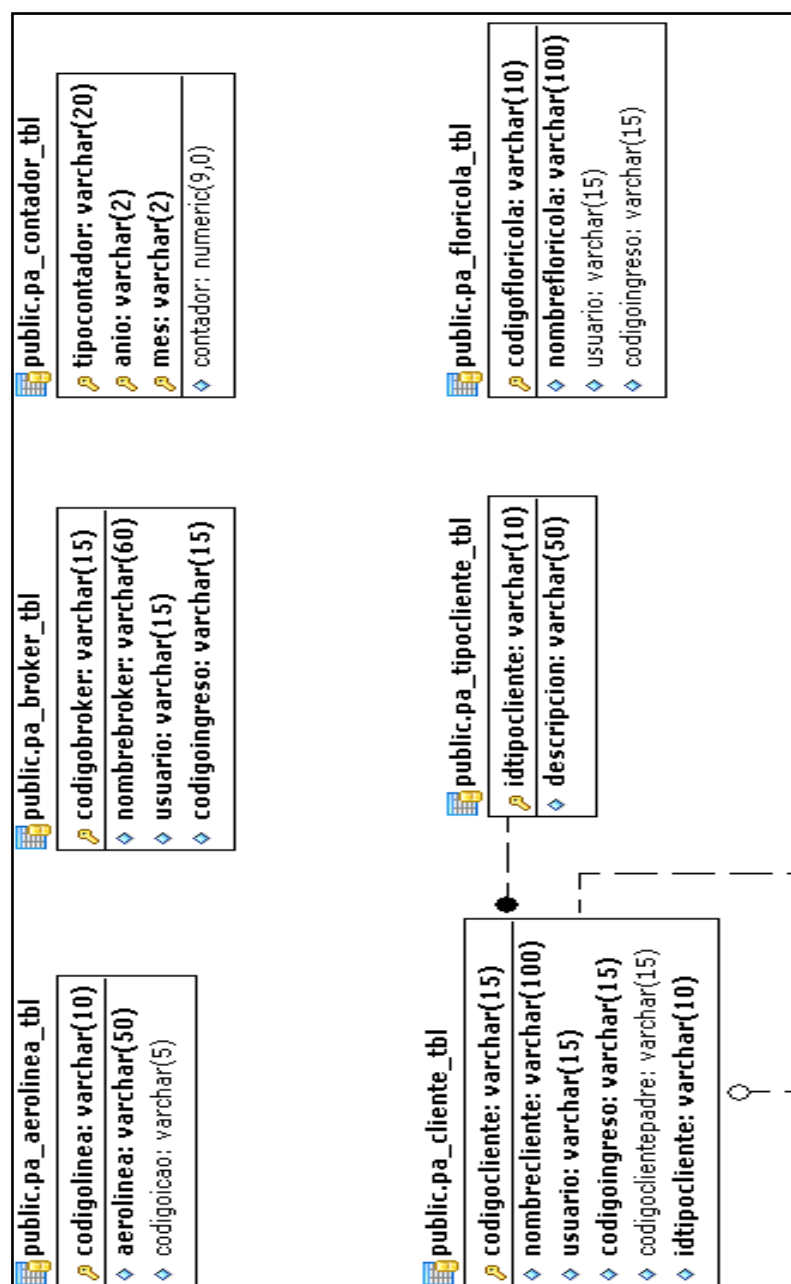


Gráfico 50. Literal b) Tablas de Parámetros.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

c. Relación awb_reservacionmensual_tbl con parámetros:

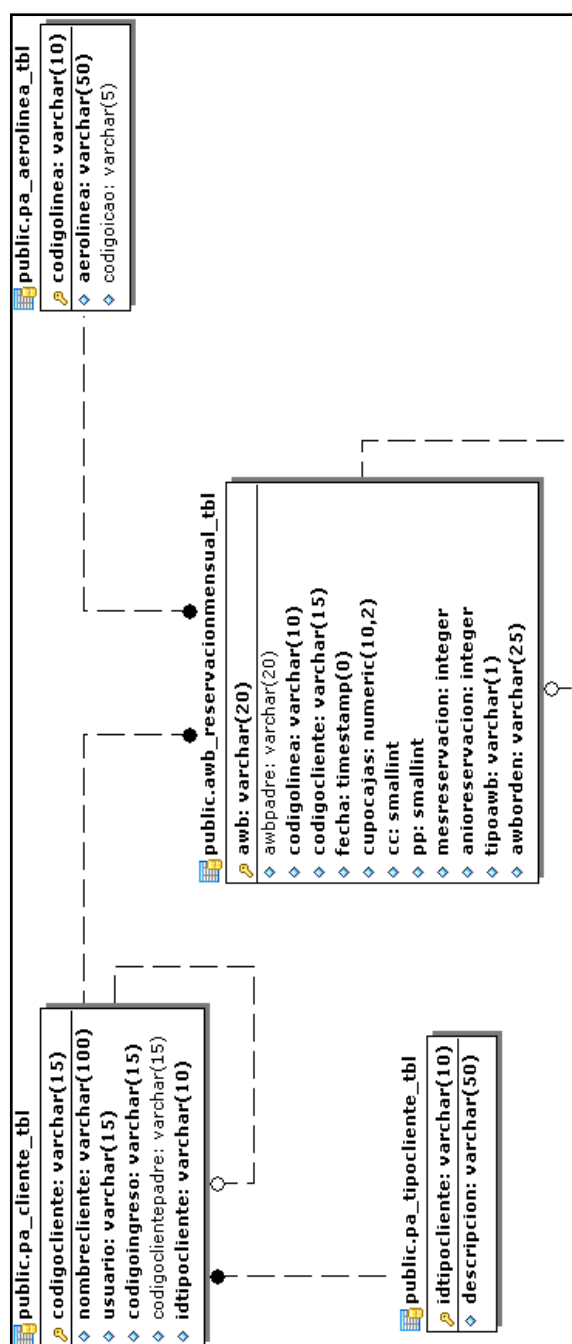


Gráfico 51. literal c) Relación awb_reservacionmensual_tbl con parámetros.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- d. Relación awb_cabeceraexportacion_tbl con parámetros:

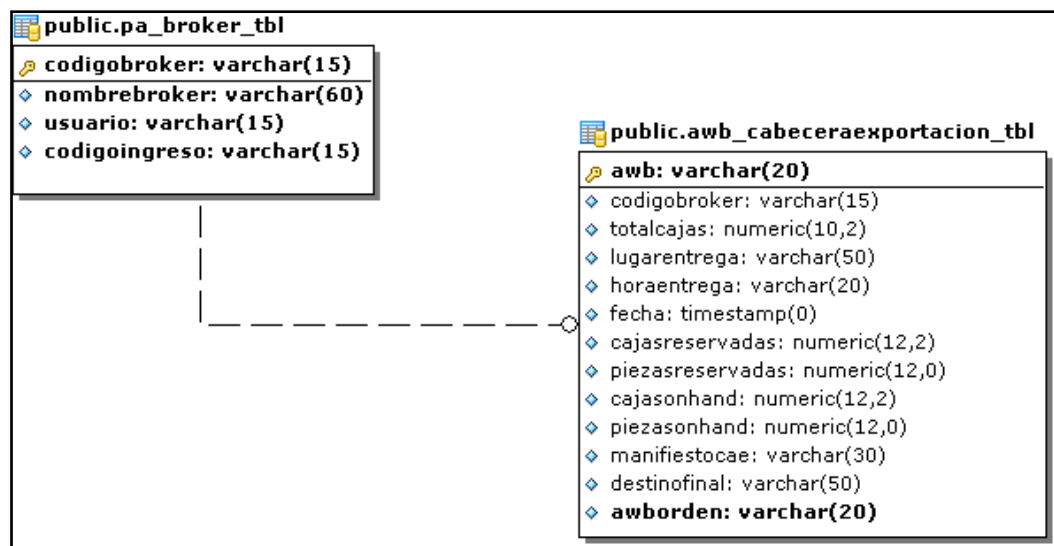


Gráfico 52. literal d) Relación awb_cabeceraexportacion_tbl con parámetros.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- e. Relación awb_detalleexportacion_tbl con parámetros:

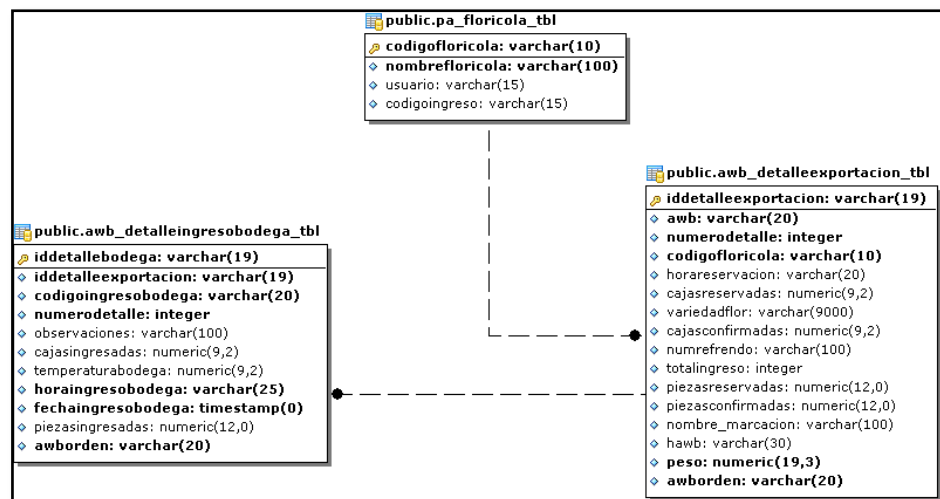


Gráfico 53. literal e) Relación awb_detalleexportacion_tbl con parámetros.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

3.2.2 DICCIONARIO DE LA BASE DE DATOS

A continuación se muestra el diccionario de la base de datos, con sus tablas, tipos de dato, y otras definiciones básicas:

- awb_reservacionmensual_tbl:

P K	F K	NAME	DATA TYPE	NOT NUL L	UNI QUE	INHE RITE D	DEFA ULT	DESCRIPTION
✓		awb	varchar(20)	✓	✓			GUÍA DEL CLIENTE
	✓	awbpadre	varchar(20)					ID AWB PADRE
	✓	codigolinea	varchar(10)	✓				CODIGO DE LA AEROLÍNEA
	✓	codigocliente	varchar(15)	✓				CODIGO DEL CLIENTE
		fecha	timestamp(0)	✓				FECHA DE RESERVACIÓN
		cupocajas	numeric(10,2)	✓				CUPO DE CAJAS
		cc	smallint	✓				FORMA PAGO CC
		pp	smallint	✓				FORMA PAGO PP
		mesreservacion	integer	✓				MES DE LA RESERVACIÓN
		anioreservacion	integer	✓				AÑO DE LA RESERVACIÓN
		tipoawb	varchar(1)	✓				TIPO DE GUÍA
		awborden	varchar(25)	✓				GUÍA DE ORDEN

Tabla 1. awb_reservacionmensual_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- awb_cabeceraexportacion_tbl:

P K	F K	NAME	DATA TYPE	NOT NUL L	UNI QU E	INHE RITE D	DEFAUL T	DESCRIPTION
✓	✓	awb	varchar(20)	✓	✓			GUÍA DEL CLIENTE
	✓	codigobroker	varchar(15)					CÓDIGO DEL BRÓKER
		totalcajas	numeric(10,2)					TOTAL DE CAJAS DE LA GUÍA
		lugarentrega	varchar(50)					LUGAR DE ENTREGA DE LA CARGA
		horaentrega	varchar(20)					HORA DE ENTREGA DE LA CARGA
		fecha	timestamp(0)					FECHA DE ENTREGA DE LA CARGA
		cajasreservadas	numeric(12,2)				0	TOTAL DE CAJAS RESERVADAS
		piezasreservadas	numeric(12,0)				0	TOTAL DE PIEZAS RESERVADAS
		cajasonhand	numeric(12,2)				0	TOTAL DE CAJAS EN CUARTO FRÍO
		piezasonhand	numeric(12,0)				0	TOTAL DE PIEZAS EN CUARTO FRÍO
		manifestocae	varchar(30)				'- ':character varying	MANIFIESTO DE LA GUÍA
		destinofinal	varchar(50)					DESTINO FINAL DE LA GUÍA
		awborden	varchar(20)	✓				AWB ORDEN DE LA CABECERA DE EXPORTACIÓN

Tabla 2. awb_cabeceraexportacion_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- awb_detalleexportacion_tbl:

P K	F K	NAME	DATA TYPE	NOT NUL L	UNI QUE	INH ERI TED	DEFAULT	DESCRIPTION
✓		iddetalleexportacion	varchar(19)	✓	✓			IDENTIFICADOR
	✓	awb	varchar(20)	✓				GUÍA DE LA FLORÍCOLA
		numerodetalle	integer	✓				NÚMERO DE DETALLE
	✓	codigofloricola	varchar(10)	✓				CÓDIGO DE LA FLORÍCOLA
		horareservacion	varchar(20)					HORA DE RESERVACIÓN
		cajasreservadas	numeric(9,2)					TOTAL DE CAJAS RESERVADAS
		variedadflor	varchar(9000)					VARIEDAD DE LA FLOR
		cajasconfirmadas	numeric(9,2)					TOTAL DE CAJAS EN CUARTO FRÍO
		numrefrendo	varchar(100)					NÚMERO DE REFRENDO
		totalingreso	integer					TOTAL DE INGRESOS
		piezasreservadas	numeric(12,0)				0	TOTAL DE PIEZAS RESERVADAS
		piezasconfirmadas	numeric(12,0)				0	TOTAL DE PIEZAS CONFIRMADAS
		nombre_marcacion	varchar(100)					NOMBRE DE LA MARCACIÓN
		hawb	varchar(30)				'':character varying	HAWB
		peso	numeric(19,3)	✓			0	PESO
		awborden	varchar(20)	✓				AWB ORDEN DEL DETALLE DE EXPORTACIÓN, COORDINACIONES DE FLORÍCOLAS

Tabla 3. awb_detalleexportacion_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- awb_detalleingresobodega_tbl:

P K	F K	Name	Data type	Not nul l	Uniq ue	Inh erit ed	Default	Description
✓		iddetallebodega	varchar(19)	✓	✓			ID PRINCIPAL DE LA TABLA
	✓	iddetalleexportacion	varchar(19)	✓				CLAVE FORANEA DE LA TABLA DETALLE DE EXPORTACIÓN
		codigoingresobodega	varchar(20)	✓			'0':charac ter varying	NÚMERO DE INGRESO DE BODEGA
		numerodetalle	integer	✓				NÚMERO DE DETALLE
		observaciones	varchar(100)					OBSEVACIONES AL INGRESO
		cajasingresadas	numeric(9,2)					CAJAS INGRESADAS
		temperaturabodega	numeric(9,2)					TEMPERATURA
		horaingresobodega	varchar(25)	✓				HORA INGRESO BODEGA
		fechaingresobodega	timestamp(0)	✓				FECHA DE INGRESO BODEGA
		piezasingresadas	numeric(12,0)				0	PIEZAS INGRESADAS
		awborden	varchar(20)	✓				AWB ORDEN DE LOS REGISTROS DE INGRESO DE BODEGA

Tabla 4. awb_detalleingresobodega_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- awb_documentoexportacion_tbl:

P K	F K	Name	Data type	Not nul l	Uniq ue	Inh erit ed	Def aul t	Description
✓		iddocumentoexportacion	numeric(19,0)	✓	✓			ID PRINCIPAL DE LA TABLA
	✓	awb	varchar(20)					NUMERO DE LA GUIA
		imagen	varchar(250)	✓				NOMBRE DE LA IMAGEN
		conceptodocumento	varchar(80)	✓				NOMBRE DEL DOCUMENTO
		broker	integer					VISUALIZACION PARA BROKER
		awborden	varchar(20)	✓				AWB ORDEN PARA EL REGISTRO DE DOCUMENTOS SUBIDOS A LA WEB

Tabla 5. awb_documentoexportacion_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- awb_rutaawb_tbl:

PK	FK	Name	Data type	Not null	Unique	Inherited	Default	Description
✓		idruta	numeric(19,0)	✓	✓			ID PRINCIPAL DE LA TABLA
	✓	awb	varchar(20)	✓				GUIA DEL VUELO
		orgaeropuerto	varchar(10)	✓				AEROPUERTO ORIGEN
		orgaerolinea	varchar(50)	✓				AEROLINEA ORIGEN
		orden	integer	✓				ORDEN DE LA RUTA
		orgfechasalida	date					FECHA SALIDA ORIGEN
		horasalidorga	varchar(20)					HORA SALIDA ORIGEN
		numvuelo	varchar(30)					NUMERO DE VUELO
		destaeropuerto	varchar(10)					AEROPUERTO DESTINO
		destfechallegada	date					FECHA LLEGADA DESTINO
		desthorallegada	varchar(20)					HORA LLEGADA DESTINO
		awborden	varchar(20)	✓				AWB ORDEN DE LAS RUTAS DE LA GUIA

Tabla 6. awb_rutaawb_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- pa_aerolinea_tbl:

PK	FK	Name	Data type	Not null	Unique	Inherited	Default	Description
✓		codigolinea	varchar(10)	✓	✓			CODIGO DE LA AEROLINEA
		aerolinea	varchar(50)	✓				NOMBRE DE LA AEROLINEA
		codigoicao	varchar(5)					INICIALES DE LA AEROLINEA

Tabla 7. pa_aerolinea_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- pa_broker_tbl:

PK	F K	Name	Data type	Not null	Uniq ue	Inherit ed	Default	Description
✓		codigobroker	varchar(15)	✓	✓			CODIGO DEL BROKER
		nombrebroker	varchar(60)	✓				NOMBRE DEL BROKER
		usuario	varchar(15)	✓				USUARIO DE INGRESO WEB
		codigoingreso	varchar(15)	✓				CODIGO DE INGRESO WEB

Tabla 8. pa_broker_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- pa_cliente_tbl

P K	F K	Name	Data type	Not null	Uniq ue	Inherited	Default	Description
✓		codigocliente	varchar(15)	✓	✓			CODIGO DEL CLIENTE
		nombreciente	varchar(100)	✓				NOMBRE DEL CLIENTE
		usuario	varchar(15)	✓				USUARIO DE INGRESO DE LA WEB
		codigoingreso	varchar(15)	✓				CODIGO DE INGRESO DE LA WEB
	✓	codigoclientepadre	varchar(15)					CODIGO DEL CLIENTE PADRE
	✓	idtipocliente	varchar(10)	✓				TIPO DE CLIENTE

Tabla 9. pa_cliente_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- pa_floricola_tbl

P K	F K	Name	Data type	Not null	Uniq ue	Inherited	Default	Description
✓		codigocliente	varchar(15)	✓	✓			CODIGO DEL CLIENTE
		nombreciente	varchar(100)	✓				NOMBRE DEL CLIENTE
		usuario	varchar(15)	✓				USUARIO DE INGRESO DE LA WEB
		codigoingreso	varchar(15)	✓				CODIGO DE INGRESO DE LA WEB
	✓	codigoclientepadre	varchar(15)					CODIGO DEL CLIENTE PADRE
	✓	idtipocliente	varchar(10)	✓				TIPO DE CLIENTE

Tabla 10. pa_floricola_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- pa_tipocliente_tbl

P K	FK	Name	Data type	Not null	Uniq ue	Inhe ritad	Defaul t	Description
✓		idtipocliente	varchar(10)	✓	✓			ID PRINCIPAL DE LA TABLA
		descripcion	varchar(50)	✓				DESCRIPCIÓN DE LA TABLA

Tabla 11. pa_tipocliente_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- pa_contador_tbl

PK	F K	Name	Data type	Not null	Uniq ue	Inherit ed	Defa ult	Description
✓		tipocontador	varchar(20)	✓				TIPO DE CONTADOR
✓		anio	varchar(2)	✓				ANIO DE INGRESO
✓		mes	varchar(2)	✓				MES DE INGRESO
		contador	numeric(9,0)					NUMERO DE CONTADOR

Tabla 12. pa_contador_tbl.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

3.3 Diseño de interfaz gráfica

3.3.1 Framework CSS

a .CSS.- Es un lenguaje como otro cualquiera, orientado a definir los parámetros esenciales en un documento HTML. Por ejemplo, con CSS se puede customizar¹⁷

¹⁷ **CUSTOMIZAR:** es un verbo que no forma parte del diccionario de la Real Academia Española (RAE) pero que, sin embargo, tiene un uso bastante frecuente en nuestra lengua. Se trata de una adaptación del término inglés customize, que refiere a modificar algo de acuerdo a las preferencias personales. Puede decirse, por lo tanto, que customizar un objeto es lo mismo que personalizarlo (adaptarlo a nuestro gusto). La noción de customizar es frecuente en el ámbito de la moda y de la indumentaria, aunque también se utiliza en el marketing.

parámetros acerca de las fuentes, colores, márgenes, líneas, imágenes de fondo, posición de contenidos, tabulaciones, anchuras, rellenos, etc.

“HTML permite desarrollar un documento básico con preformato. Con CSS se puede definir todo el formato de una manera sofisticada y personalizada. CSS es el código que leen todos los navegadores actuales, aunque cada uno a su debida manera. Los navegadores hacen interpretaciones del CSS y recrean su lectura en el sitio web donde se procesa” (Tecnoquo, 2009).

b. Diferencia entre HTML y CSS.- HTML es utilizado para estructurar los contenidos, es el cuerpo que contiene la información. CSS es usado para formatear contenidos estructurados.

c. Ventajas de CSS.- Entre las principales ventajas se puede determinar: Control de varios documentos a partir de un mismo fichero CSS:

- Mayor precisión y facilidad de edición ya que todos se vinculan al mismo fichero patrón;
- Aplicación de diferentes “layouts” para servir a diferentes medidas: resoluciones de pantalla, impresora, soportes digitales, etc; y,
- Empleo de sofisticadas técnicas de desarrollo.

d. Framework.- En el desarrollo de software, un framework es una estructura conceptual y tecnológica de soporte definida, normalmente con artefactos o módulos de software concretos, con base en la cual otro proyecto de software puede ser organizado y desarrollado.

e. Componentes RichFaces.- Es una implementación JSF basada en AJAX.

“Es posible crear componentes propios y ponerlos en práctica mediante funcionalidad AJAX, sin embargo, esto podría hacer que se tomará demasiado tiempo en el desarrollo” (VILLANUEVA, 2008).

Estas implementaciones proporcionan facilidad de desarrollo y reutilización; es por ello por lo que muchos desarrolladores se decantan por este tipo de herramientas.

Existen diversas implementaciones de JSF basadas en AJAX, como son Ajax4JSF, TomaHawk, RichFaces, Trinidad, IceFaces, y muchos otros.

No obstante, cada una es diferente a otra. Dentro de este conjunto, las implementaciones Ajax4JSF, RichFaces y IceFaces son las que cobran mayor importancia debido a su divulgación entre la comunidad web.

Rich Faces es un framework de código abierto que añade capacidad Ajax dentro de aplicaciones JSF existentes sin recurrir a JavaScript. RichFaces incluye ciclo de vida, validaciones, conversores y la gestión de recursos estáticos y dinámicos.

Los componentes de RichFaces están contruidos con soporte Ajax y un alto grado de personalización del “look-and-feel” que puede ser fácilmente incorporado dentro de las aplicaciones JSF.

Características.

- Intensificar el conjunto de beneficios JSF al trabajar con Ajax. Rich Faces está completamente integrado en el ciclo de vida de JSF. Mientras que otros marcos sólo dan acceso a los managed bean. RichFaces permite acceder al action y al valor del listener;

- Añade capacidad Ajax a aplicaciones JSF. El framework proporciona dos librerías de componentes (Core Ajax y la interfaz de usuario). La librería Core permite agregar la funcionalidad Ajax en las páginas que queramos sin necesidad de escribir nada de código JavaScript. Rich Faces permite definir eventos en la propia página. Un evento invoca a una petición Ajax, sincronizándose así zonas de la página y componentes JSF después de recibir la respuesta del servidor por Ajax;
- Crear rápidamente vistas complejas basándose en la caja de componentes. La librería UI (Interfaz de usuario) que contiene componentes para agregar características de interfaz de usuario a aplicaciones JSF. Se amplía el framework de RichFaces incluyendo un gran conjunto de componentes “habilitación de Ajax” que extiende el soporte de la página. Además, los componentes de RichFaces están diseñados para ser usados sin problemas con otras librerías de componentes en la misma página, de modo que existen más opciones para el desarrollo de aplicaciones;
- Proporciona un paquete de recursos con clases de aplicación JAVA. Además de su núcleo, la funcionalidad de RichFaces para Ajax proporciona un avanzado soporte a la gestión de diferentes recursos: imágenes, código JavaScript y hojas de estilo CSS. El framework de recursos hace posible empaquetar fácilmente estos recursos en archivos jar junto con el código de los componentes personalizados; y,
- Generar fácilmente recursos binarios sobre la marcha. Los recursos del framework pueden generar imágenes, sonidos, hojas de cálculo de Excel, etc.

Los componentes de la interfaz de usuario de RichFaces vienen preparados para su uso fuera del paquete, así los desarrolladores ahorrarán tiempo y podrán disponer de las ventajas mencionadas para la creación de aplicaciones Web. Como resultado, la experiencia puede ser más rápida y fácil de obtener.

El esquema de la solicitud de procesamiento de flujo sería el que se muestra en la siguiente figura:

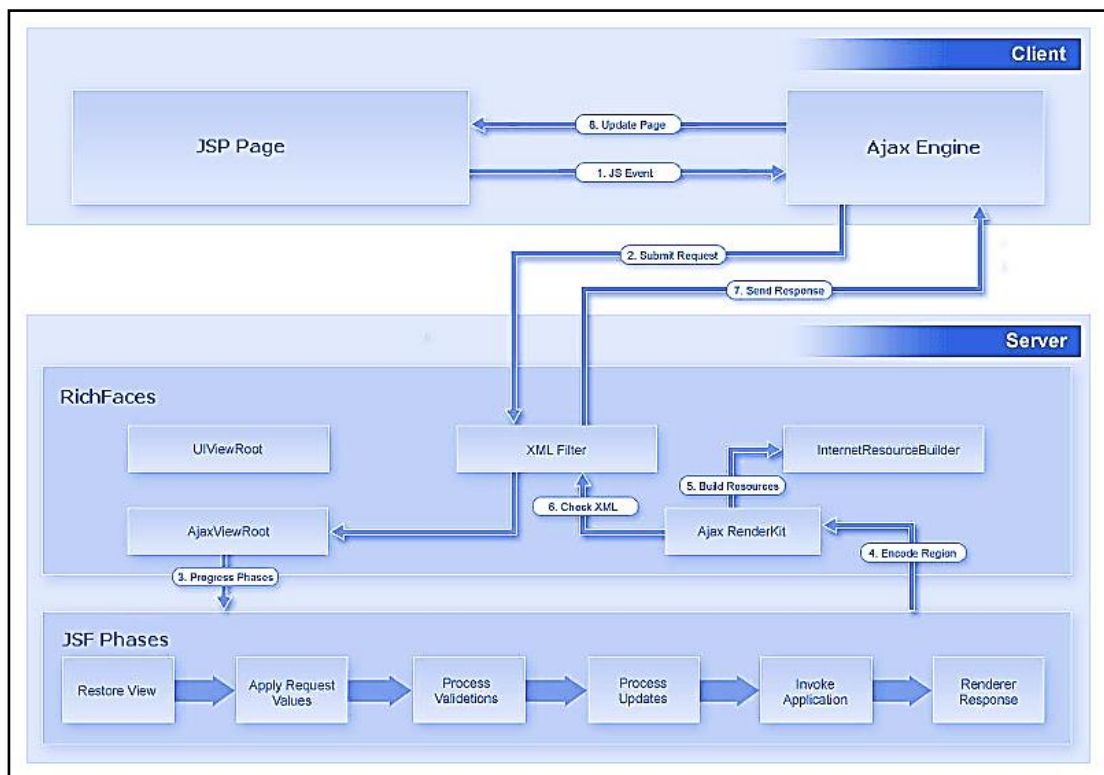


Gráfico 54. Esquema de solicitud de procesamiento de flujo.

Fuente: <http://www.juntadeandalucia.es/xwiki/bin/view/MADEJA/RichFaces>.

Autor: Junta de Andalucía.

Entre los elementos que componen RichFaces se encuentran:

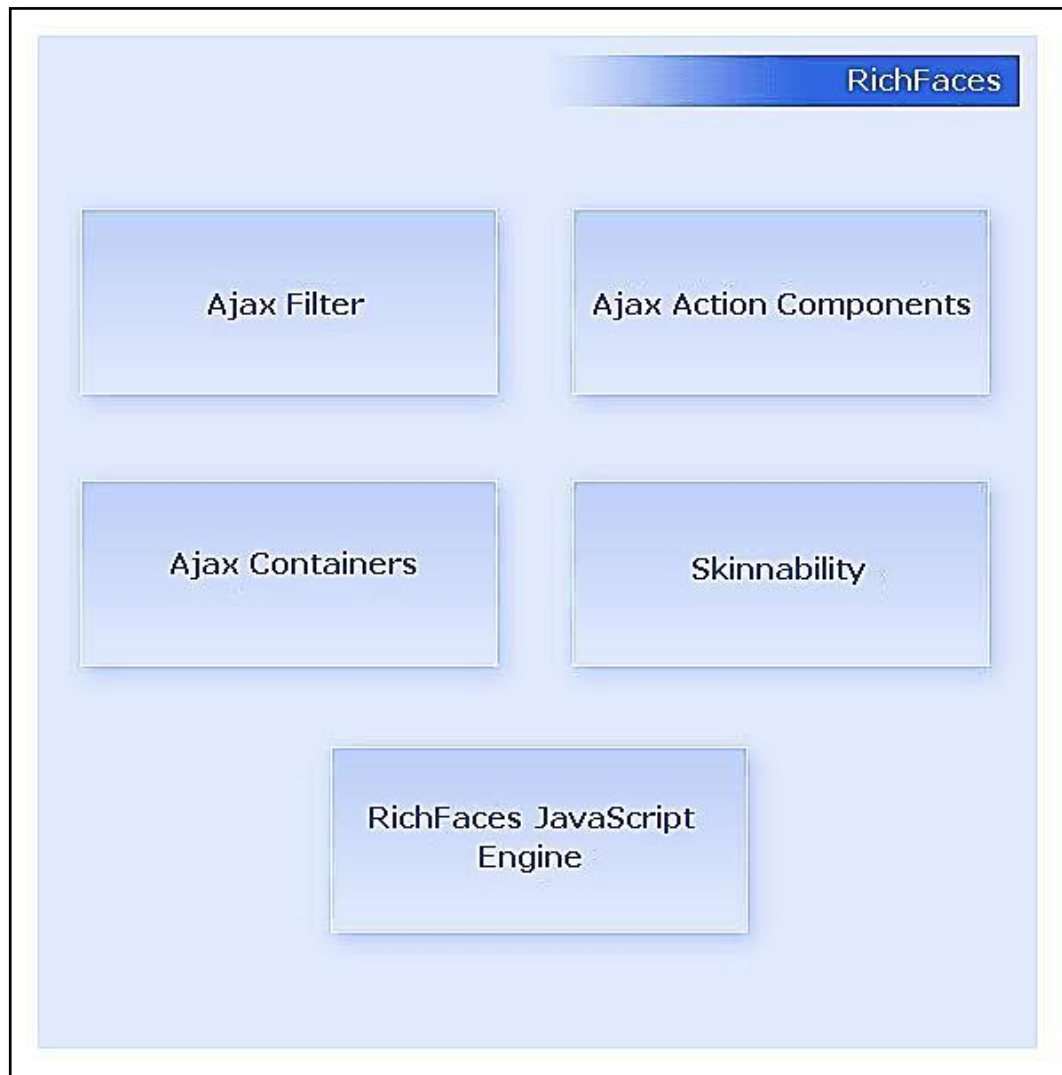


Gráfico 55. Elementos RichFaces.

Fuente: <http://www.juntadeandalucia.es/xwiki/bin/view/MADEJA/RichFaces>

Autor: Junta de Andalucía.

- Ventajas:

- Al pertenecer RichFaces a un subproyecto de JBoss, su integración con Seam es perfecta; y,
- Al ser RichFaces propiedad de Exadel, se ajusta perfectamente al IDE Red Hat Developer Studio

- Inconvenientes

- No se puede realizar una aplicación combinándolo con IceFaces y Seam.

a. **Filtros en Aplicaciones Web.**- La idea de un filtro es la de interceptar un request o response, agregando una acción, lógica o función. Las actividades básicas que estos realizan son:

- Permitir y bloquear acceso;
- Caché, compresión, logging;
- Autenticación, control de acceso, encriptación; y,
- Transformación de contenido.

La versión 2.3 de la especificación de los Servlets introduce los Filtros como un medio flexible de interactuar con peticiones y respuestas HTTP antes y después de que el Contenedor Web llame a los recursos Web. Estos recursos Web pueden ser tanto estáticos como dinámicos:

- Recursos Web estáticos: páginas HTML e imágenes; y,
- Recursos Web dinámicos: Servlets y páginas JSP

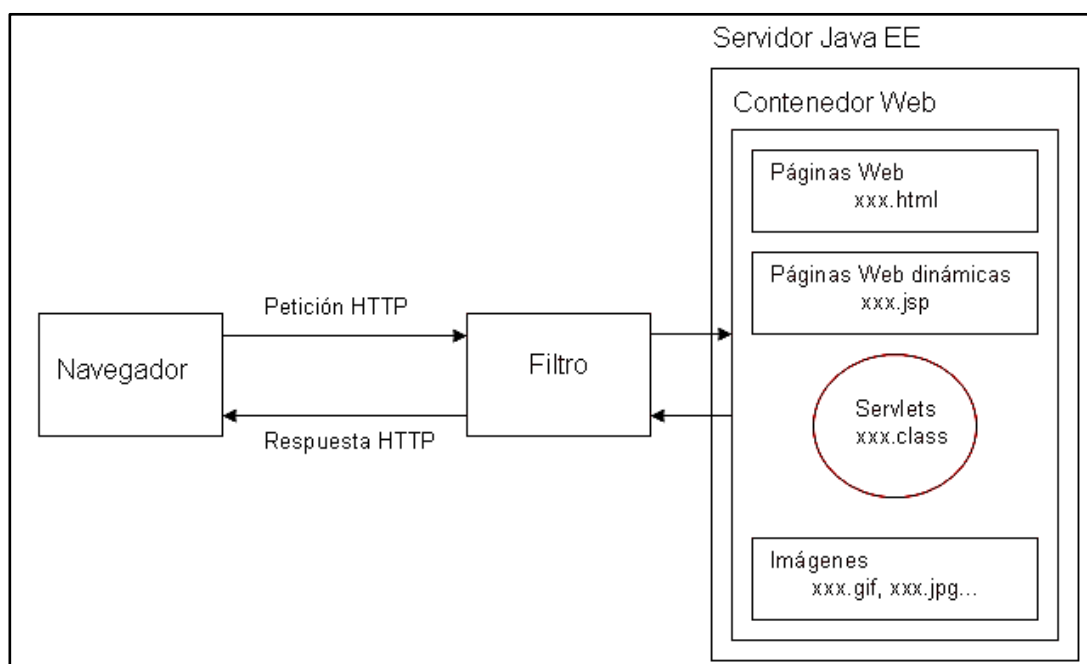


Gráfico 56. Filtros en Aplicación Web.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

- Usos principales:

- i. Encapsular tareas comunes en unidades reutilizables;
- ii. Si por ejemplo un Servlet que realiza las funciones de fachada tiene que encargarse de ejecutar 6 acciones en un determinado orden, estas seis acciones se pueden escribir en 6 Filtros independientes e indicar su orden en el fichero descriptor de despliegue de la aplicación **web.xml**;
- iii. Si en el futuro otra aplicación necesitara por ejemplo 4 de esos 6 filtros y en diferente orden, lo único que hace es indicarle al fichero descriptor de la nueva aplicación que tiene que trabajar con 4 Filtros que ya tiene otro proyecto e indicarle su correspondiente orden en el mencionado fichero;
- iv. Realizar labores de autenticación y bloqueo de peticiones;
- v. Realizar estadísticas de impactos al Sitio Web; y,
- vi. Formatear los datos enviados al usuario.

1. Los datos enviados como respuesta al usuario pueden ser transformados al pasar por un Filtro de salida. En este caso los Filtros pueden:

- a. Comprimir la información;
- b. Manipular imágenes para sobreponer en ellas un texto o logo representativo del sitio Web; y,
- c. Encriptar información.

3.3.2 Framework CSS del sistema

El framework utilizado para el sistema es:

CSS	DESCRIPCIÓN
layout.css	CSS base que especifica el cuerpo de la página web en: Cabecera, Barra de menú, pie de página, y contenido.
layout-without-session.css	CSS aplicado a la aplicación cuando el usuario no se encuentra todavía autenticado en la aplicación.
layout-without-nav.css	CSS aplicado cuando el usuario se encuentra autenticado, pero no despliega la barra de menú.
layout-nav-left.css	CSS aplicado cuando el usuario se encuentra autenticado y despliega la barra de menú principal.
layout-nav-left-hide.css	CSS aplicado cuando el usuario se encuentra autenticado y despliega la barra de menú principal, con opción para ocultar o habilitar el menú.
message.css	CSS para la presentación de mensajes de: OK, Warning, Info, Error.
typography.css	CSS que especifica el tipo de letra para títulos, párrafos, links.
tables.css	CSS para la definición de las tablas de uso en el sistema.
util.css	CSS utilitario para especificar los tipos de segmentación de la etiqueta div.

Tabla 13. Framework utilizado para el sistema.

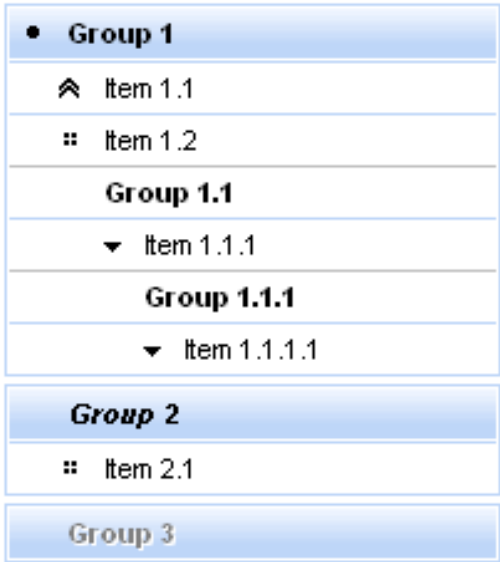

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

3.3.3 Componentes RichFaces

“RichFaces es un framework de código abierto que añade capacidad Ajax dentro de aplicaciones JSF existentes sin recurrir a JavaScript. RichFaces incluye ciclo de vida, validaciones, conversores y la gestión de recursos estáticos y dinámicos” (RODRIGUÉZ, 2012).

Componentes utilizados:

COMPONENTE	DESCRIPCIÓN
rich:panelMenu	<p>Maneja el menú de opciones de cada usuario:</p>  <p>The screenshot shows a panel menu with three main groups: 'Group 1', 'Group 2', and 'Group 3'. 'Group 1' is expanded, showing 'Item 1.1' (with a right-pointing triangle icon) and 'Item 1.2' (with a double hash icon). 'Item 1.1' is further expanded to show 'Item 1.1.1' (with a downward-pointing triangle icon), which is also expanded to show 'Item 1.1.1.1' (with a downward-pointing triangle icon). 'Group 2' contains 'Item 2.1' (with a double hash icon). 'Group 3' is currently collapsed.</p>
a4j:commandButton	<p>Botones de acciones para las opciones</p>  <p>The screenshot shows a rectangular button with a light blue gradient and a thin border, containing the text 'Do something' in a dark blue font.</p>
rich:calendar	<p>Selección de fechas.</p>

















































	<div><div>Aug 30, 2007</div><div><div>→</div><div>Aug 30, 2007</div></div><div><div>optionalHeader Facet</div><div><< < August, 07 > >></div><div>Sun Mon Tue Wed Thu Fri Sat</div><div><div>31 29 30 31 1 2 3 4</div><div>32 5 6 7 8 9 10 11</div><div>33 12 13 14 15 16 17 18</div><div>34 19 20 21 22 23 24 25</div><div>35 26 27 28 29 30 31 1</div><div>36 2 3 4 5 6 7 8</div></div><div>Aug 30, 2007 (x) Today</div><div>optionalFooter Facet</div></div></div>																																																			
rich:scrollableDataTable	<p>Desplegar datos en un grid que permite tener un scroll.</p> <table><thead><tr><th>State</th><th>Flag</th><th>Capital</th></tr></thead><tbody><tr><td>Alabama</td><td></td><td>Montgomery</td></tr><tr><td>Alaska</td><td></td><td>Juneau</td></tr><tr><td>Arizona</td><td></td><td>Phoenix</td></tr><tr><td>Arkansas</td><td></td><td>Little Rock</td></tr><tr><td>California</td><td></td><td>Sacramento</td></tr><tr><td>Colorado</td><td></td><td>Denver</td></tr><tr><td>Connecticut</td><td></td><td>Hartford</td></tr><tr><td>Delaware</td><td></td><td>Dover</td></tr><tr><td>Florida</td><td></td><td>Tallahassee</td></tr><tr><td>Georgia</td><td></td><td>Atlanta</td></tr><tr><td>Hawaii</td><td></td><td>Honolulu</td></tr><tr><td>Idaho</td><td></td><td>Boise</td></tr><tr><td>Illinois</td><td></td><td>Springfield</td></tr><tr><td>Iowa</td><td></td><td>Des Moines</td></tr><tr><td>Kansas</td><td></td><td>Topeka</td></tr><tr><td>Kentucky</td><td></td><td>Frankfort</td></tr></tbody></table>	State	Flag	Capital	Alabama		Montgomery	Alaska		Juneau	Arizona		Phoenix	Arkansas		Little Rock	California		Sacramento	Colorado		Denver	Connecticut		Hartford	Delaware		Dover	Florida		Tallahassee	Georgia		Atlanta	Hawaii		Honolulu	Idaho		Boise	Illinois		Springfield	Iowa		Des Moines	Kansas		Topeka	Kentucky		Frankfort
State	Flag	Capital																																																		
Alabama		Montgomery																																																		
Alaska		Juneau																																																		
Arizona		Phoenix																																																		
Arkansas		Little Rock																																																		
California		Sacramento																																																		
Colorado		Denver																																																		
Connecticut		Hartford																																																		
Delaware		Dover																																																		
Florida		Tallahassee																																																		
Georgia		Atlanta																																																		
Hawaii		Honolulu																																																		
Idaho		Boise																																																		
Illinois		Springfield																																																		
Iowa		Des Moines																																																		
Kansas		Topeka																																																		
Kentucky		Frankfort																																																		
a4j:form	Manejo del elemento <form> para utilizar Ajax en los componentes contenedores.																																																			
a4j:status	Componente que indica el estado de la acción realizada por el usuario.																																																			

Tabla 14. Componentes utilizados.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

Librerías para Rich Faces 3.3.0 Final:

- common-annotations.jar;
- commons-beanutils.jar;
- commons-collections.jar;
- commons-digester.jar;
- commons-discovery-0.2.jar;
- commons-entico-1.0.jar;
- commons-logging.jar;
- richfaces-api-3.3.0.GA.jar;
- richfaces-impl-3.3.0.GA.jar;
- richfaces-ui-3.3.0.GA.jar;
- darkX-3.3.0.GA.jar;
- jsf-api.jar;
- jsf-impl.jar;
- laguna-3.3.0.GA.jar;
- spring.jar; y,
- wsdl4j.jar.

3.3.4 Templates Facelets

JavaServer Facelets es un framework para plantillas (templates) centrado en la tecnología JSF (JavaServer Faces), por lo cual se integran de manera muy fácil.

Este framework incluye muchas características siendo las más importantes:

- Tiempo de desarrollo cero de los tags para UIComponents;
- Facilidad en la creación del templating para los componentes y páginas;
- Habilidad de separar los UIComponents en diferentes archivos;
- Un buen sistema de reporte de errores;
- Soporte completo a EL (Expression Language;

- Validación de EL en tiempo de construcción;
- No es necesaria configuración XML; y,
- Trabaja con cualquier RenderKit.

Las plantillas utilizadas para todas las páginas JSF son:

PLANTILLA	DESCRIPCIÓN
baseSinSessionPrincipal.xhtml	Template base utilizada para todas las páginas JSF que no se encuentren en la sesión autenticada por el usuario.
baseConSessionPrincipal.xhtml	Template base utilizada para todas las páginas que tienen la autenticación del usuario pero que no contiene menú, se utiliza para las páginas que presentan reportes.
baseConSessionMenuPrincipal.xhtml	Template base para todas las páginas JSF que tienen autenticación por el usuario, con la característica que presenta el menú de opciones de acceso.

Tabla 15. Plantillas utilizadas para páginas JSF.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

Librerías JAVA

- jsf-facelets.jar
- jstl.jar

b. Manejo de Filtros

Servlet

Todo el filtro se maneja a través del servlet AccesActionFilter.java

Web.xml

El filtro se configura:

```

<filter>
  <filter-name>AccesFilter</filter-name>
  <filter-class>ec.com.entico.agencia.web.actions.AccesActionFilter</filter-class>
  <init-param>
    <param-name>paginasNoProtegidas</param-name>
    <param-
value>index.html,index.jsp,main.jsf,salir.jsf,sinPermisos.jsf,login.jsf,login-
error.jsf,error404.jsf,proxyAccess.do,selMod.do</param-value>
  </init-param>
</filter>

```

Tabla 16. Configuración filtro.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

CAPÍTULO IV.

SEGURIDADES

4.1. Análisis de autenticación y autorización

El acceso web por parte de los usuarios a la base de datos, está clasificado en base a los 3 perfiles que se maneja dentro del sistema; cada uno de los clientes cuenta con un usuario y una contraseña que está almacenada dentro de la base de datos; de tal manera que cuando éste quiere ingresar a la web, el sistema compara los datos de autenticación del usuario contra la base de datos y verifica si son correctos, para así ingresar a la revisión de los documentos.

Este proceso de autenticación es de vital importancia, para que el sistema pueda verificar los documentos que se deben mostrar únicamente al usuario ingresado.

- a. **Clientes:** Para que los clientes puedan acceder al sistema, éstos cuentan con un usuario y contraseña que se les proporciona en el sistema cliente – servidor dentro de la agencia.
- b. **Brókeres:** Para que los brókeres puedan acceder al sistema, éstos cuentan con un usuario y contraseña que se les proporciona en el sistema cliente – servidor dentro de la agencia.
- c. **Florícolas:** Para que los clientes puedan acceder al sistema, éstos cuentan con un usuario y contraseña que se les proporciona en el sistema cliente – servidor dentro de la agencia.

4.2. Aplicación de JAAS

“JAAS (Java Authentication and Authorization Service), es una interfaz que permite a las aplicaciones Java acceder a servicios de control de autenticación y Access” (Manuel, 2012).

Puede usarse con dos fines:

- La autenticación de usuarios: para conocer quién está ejecutando el código java; y,
- La autorización de usuarios: para garantizar quién lo ejecuta y tiene los permisos necesarios para hacerlo.

La tarea se limita a la parametrización de ficheros de configuración (*.xml): del lado del servidor de aplicaciones y del lado de la aplicación web.

El objetivo es delegar el mecanismo de autenticación y autorización en el servidor de aplicaciones. En la práctica significa, que ya no será necesario implementar un filtro de autorización en todas nuestras aplicaciones web y no se tiene que copiar el `LoginAction.java` o el `LoginBean.java` de una aplicación a otra arrastrando posibles errores.

Al implementar una interfaz que forma parte del estándar, se benefician no sólo del acceso a la información del usuario y su perfil a través de métodos que forman parte del mismo (del estándar), sino que, además, se favorece de otras implementaciones que encapsulan dichos métodos dentro de su librería de tags.

El filtro de autorización de JAAS funciona de modo que, si se accede a una página protegida y no hay usuario en la sesión, se redirecciona a la página de login. Teniendo en cuenta esto, lo que debe hacer la página de inicio es redirigir la petición a una página protegida, para que salte la página de login y tras un login satisfactorio redirija a la página protegida.” (Manuel, 2012)

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<ui:composition xmlns="http://www.w3.org/1999/xhtml"
    xmlns:ui="http://java.sun.com/jsf/facelets"
    xmlns:h="http://java.sun.com/jsf/html"
    xmlns:f="http://java.sun.com/jsf/core"
    xmlns:rich="http://richfaces.ajax4jsf.org/rich"
    xmlns:a4j="https://ajax4jsf.dev.java.net/ajax"
    template="templates/baseSinSessionPrincipal.xhtml"
    xmlns:c="http://java.sun.com/jsp/jstl/core">

    <ui:define name="titulo">
        ${titulos['sistema.titulo']}
    </ui:define>

    <ui:define name="header">

        <style type="text/css" media="screen"> @media screen {
        </style>

    </ui:define>

    <ui:define name="contenido">

        <form name="form1" method="post" action="proxyAccess.do">
        <f:subview rendered="${moduloBean.modulo=='CLI'}">
            <h2><h:outputText value="${titulos['login.modulo.cliente']}" />
            </h2>
        </f:subview> <f:subview rendered="${moduloBean.modulo=='BRO'}">
            <h2><h:outputText value="${titulos['login.modulo.broker']}" />
            </h2>
        </f:subview> <f:subview rendered="${moduloBean.modulo=='FLO'}">
            <h2><h:outputText value="${titulos['login.modulo.floricola']}" />
            </h2>
        </f:subview>
        <div class="login">
        <div class="autenticacion">
        <div class="usuarioLabel"><strong>Login :</strong></div>
        <div class="usuarioInput"><input type="text" maxlength="20"
            size="30" class="text" name="remoteUser" /></div>
        <br />
        <div class="passwordLabel"><strong>Password :</strong></div>
        <div class="passwordInput"><input type="password" maxlength="20"
            size="30" class="text" name="claveAcceso" /></div>
        <div class="autenticacionBotones" align="center">
            <f:subview rendered="${moduloBean.modulo=='CLI' or moduloBean.modulo=='BRO'}">
                <input type="submit" value="Submit" />
                <input type="reset" value="Reset" />
            </f:subview>
        </div>
    </form>
    </ui:define>
</ui:composition>

```

Gráfico 57. Archivo de Configuración: login.xhtml.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

4.3. Manejo de filtros

Para entender el manejo de los filtros es necesario entender los siguientes conceptos:

4.3.1 MVC ("Model-View-Controller")

La estructura MVC ("Model-View-Controller") es un paradigma utilizado en diversos desarrollos de software, a través de este "Framework" se logra una división de las diferentes partes que conforman una aplicación, siendo su principal razón de ser: manutención del código fuente.

Conforme incrementan las necesidades de cualquier aplicación, la modificación a código existente se hace inminente y si no existe una clara división de uso, el código no solo se torna indescifrable sino en ocasiones impredecible debido a la mezcla de funcionalidades que pueden surgir.

A través de MVC se realiza la siguiente división:

- **Model:** Concentra las funcionalidades relacionadas con el modelo de datos, esto es, el acceso y manipulación de depósitos informativos como Bases de Datos y Archivos;
- **View:** Se basa en el aspecto visual/gráfico que será empleado por la aplicación en cuestión; y,
- **Controller:** Empleado como un mediador entre el medio gráfico ("View") y el modelo ("Model"), coordina las acciones que son llevadas acabo entre ambos.

4.3.2 Con JSP'S y SERVLETS

El uso de "MVC" en ambientes Web para JSP's y Servlets ha empezado a generar gran interés, debido a que una vez diseñada una aplicación para ambiente

Web es raro que ésta permanezca sin cambios, el uso de MVC permite realizar diseños con JSP's/Servlets que logran verdaderas soluciones a escala.

En el proceso de registro que después de varios meses requiere ser modificado para solicitar datos no contemplados en el diseño inicial, o bien, la modificación de un proceso de trabajo ("Work-Flow") que requiere modificar el orden de solicitud de datos; cual sea el caso se requiere forzosamente modificación al código de JSP's y/o Servlets, observemos el siguiente gráfico con posibles resultados de una aplicación con JSP's/Servlets:

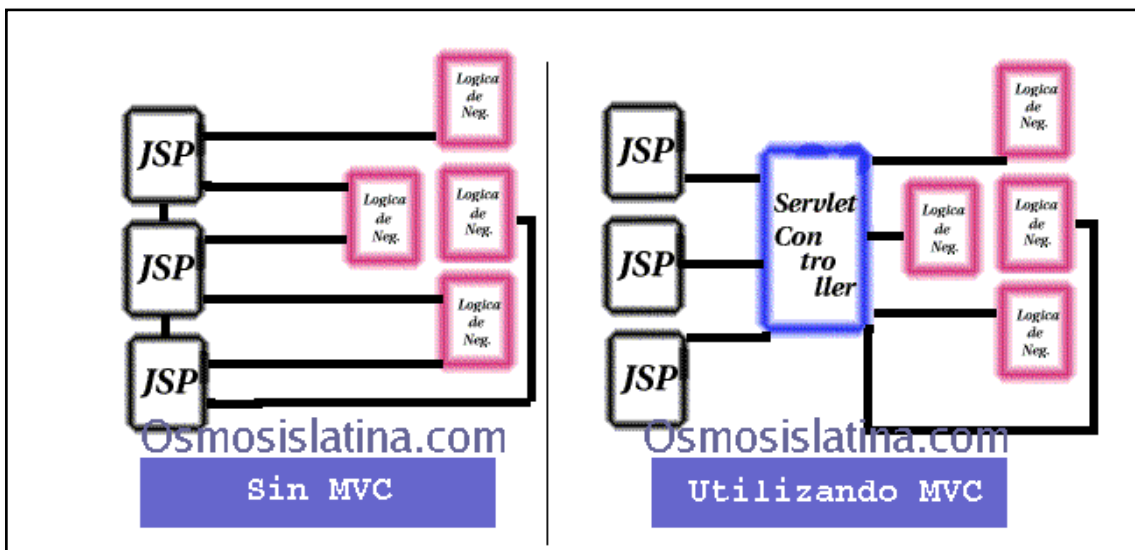


Gráfico 58. Modificación al código de JSP's y/o Servlets.

Fuente: <http://javaweb.osmosislatina.com/curso/mvc.htm#filtros>

Autor: Osmia Latina.

La sección izquierda demuestra un posible diseño que pudo ser empleado al diseñar la aplicación antes mencionada, las principales deficiencias son las siguientes:

- La secuencia de JSP's que conforma la parte visual de la aplicación esta enlazada directamente entre sí, esto es, el "WorkFlow" (Flujo de Trabajo) esta implementado en cada JSP y cualquier cambio en esta secuencia implica modificar el código fuente de diversos JSP's; y,

- “La lógica de negocios, que correspondería al Modelo de la aplicación, también esta enlazada directamente con JSP's, esto tiene como consecuencia que el diseño de lógica de negocios (a través de Servlets) esta fuertemente basado en el diseño de los JSP's frontales, lo cual obviamente dificulta que sean reutilizados diversos elementos de lógica de negocios (Servlets) en distintos JSP's” (Latina, 2012).

Utilizando "MVC" se obtienen los siguientes resultados a los problemas anteriores:

- La secuencia de JSP's es enlazada a un "Controlador" en forma de un Servlet, este enlace permite alterar el "Workflow" (Flujo de Trabajo) a través de una modificación sencilla al Servlet "Controlador", los JSP's no contienen código de "Workflow" (Flujo de Trabajo); y,

- La lógica de negocios que correspondería al Modelo es obtenida a través del Servlet "Controlador", esto permite que la misma (Servlets) permanezca aislada de cualquier tipo de despliegue gráfico (JSP's), en efecto, facilitando la reutilización de componentes de negocios (Servlets) con diversos JSP's.

Finalmente, vale mencionar que hoy en día existen diversas implementaciones para utilizar un "Framework MVC" en ambientes de JSP's/Servlets, entre ellas se encuentran:

- Struts;
- Spring; y,
- Maverick

4.3.3 Java Beans

Un Java Bean es una manera de modularizar el uso de los datos en una aplicación con JSP's/Servlets a través de una clase, su característica primordial es el uso de los métodos get y set los cuales permiten el acceso a los valores del Bean, el

diseño que utiliza un Java Bean es relativamente sencillo ya que no posee un código extenso que es utilizado para manipular datos de usuarios:

```
public class Usuario implements java.io.Serializable
{
    private String nombre;
    private String apellido;
    private String correo;

    public Usuario() {
    }

    public Usuario(String nombre, String apellido, String correo)
    {
        this.nombre = nombre;
        this.apellido = apellido;
        this.correo = correo;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }
    public String getNombre() {
        return nombre;
    }

    public void setApellido(String apellido) {
        this.apellido = apellido;
    }
    public String getApellido() {
        return apellido;
    }

    public void setCorreo(String correo) {
        this.correo = correo;
    }

    public String getCorreo(){
        return correo;
    }
}
```

Gráfico 59. Java Bean.

Fuente: <http://javaweb.osmosislatina.com/curso/mvc.htm#filtros>

Autor: Osmia Latina.

Los detalles de la clase (Java Bean) anterior son los siguientes:

- La clase es llamada Usuario e implementa la clase java.io.Serializable una característica primordial de todo Java Bean;
- Posteriormente se definen los campos ("fields") utilizados dentro del Java Bean en un contexto privado (private);

- Se definen dos Constructores Java, uno que asigna los parámetros a la instancia con valores de entrada, y el Constructor "default" sin datos de entrada; y,
- A través de los diversos métodos get/set es posible modificar los valores iniciales definidos en el Java Bean.

A través de Java Beans es posible encapsular conceptos de diferentes tipos en el proceso, permitiendo la manipulación de valores a través de una sola Clase, de esta manera facilitando el manejo de información y a su vez el desarrollo de Software; a continuación se describe el uso de Java Beans en conjunción de JSP's y Servlets:

- Java Bean Producto.java; y,
- Uso de Java Beans en JSP's/Servlets.

A simple vista el uso de Java Beans parece superficial, pero conforme va aumentando el tamaño de una aplicación su uso se hace más evidente, lo cual será ejemplificado en la siguiente sección utilizando el paradigma "MVC"("Model-View-Controller") a través de "Struts" y posteriormente en conjunción de JSTL.

4.3.4 Filtros

Otro paradigma empleado en diseños modulares de JSP's y Servlets es el de Filtros, como su nombre lo implica, un filtro permite que determinada requisición o respuesta sea analizada en circunstancias específicas. En este caso la requisición y/o respuesta corresponden a los principales elementos de un Servlet (por ende JSP también) que son los objetos ServletRequest y ServletResponse respectivamente.

Una de las principales características de un filtro que éste puede ser colocado en cualquier punto de una secuencia de actividades ("Work-Flow"), ya sea entre una solicitud de JSP a JSP, Servlet a Servlet, o inclusive Servlet/JSP a HTML, la única condición a cumplirse es que debe existir un objeto de entrada (ServletRequest) y otro de salida (ServletResponse), condición que obviamente cumple todo JSP y Servlet.

Ahora bien, en cuanto a lo que concierne el proceso de filtrado, éste puede ser de cualquier tipo imaginable, aunque los candidatos naturales para filtros son procesos de registro ("Logs"), pruebas de integridad ("Debugging") o simplemente la modificación de una respuesta previo a su envío. (Latina, Menú Curso Java Web (JSP's/Servlets), 2011).

4.3.4.1 Definición de un filtro en *web.xml*

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<!DOCTYPE web-app
PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">

<web-app>

    <filter>
        <filter-name>ImpuestosRegistro</filter-name>
        <filter-class>com.osmosislatina.filtros.Impuestos
        <init-param>
            <param-name>impuestos</param-name>
            <param-value>15</param-value>
        </init-param>
    </filter>

    <filter-mapping>
        <filter-name>ImpuestosRegistro</filter-name>
        <url-pattern>/javabeans/subasta.jsp</url-pattern>
    </filter-mapping>

    <filter-mapping>
        <filter-name>ImpuestosRegistro</filter-name>
        <url-pattern>/javabeans/subasta_cambio.jsp</url-pattern>
    </filter-mapping>

</web-app>
```

Gráfico 60. Definición de un filtro en web.xml

Fuente: http://javaweb.osmosislatina.com/curso/filtros/webxml_filtros.htm

Autor: Osmia Latina.

- `<filter>` es el elemento raíz empleado para definir un filtro, dentro de este elemento se anidan `<filter-name>` empleado para definir un nombre descriptivo del filtro y `<filter-class>` que indica la clase que contiene el filtro (Dicha clase también sería colocada dentro del directorio `clases`, o bien , dentro de un archivo JAR colocado en el directorio `lib`, como indica la estructura de WAR's);

- Además de estos últimos elementos, también pueden ser agregados los elementos `<init-param>` para indicar elementos de configuración iniciales del filtro; en caso de ser proporcionados dichos valores, cada uno debe incluir un par de elementos : `<param-name>` y `<param-value>` para especificar el nombre y valor del parámetro respectivamente;

- Una vez definido un filtro, se indica un mapeo* para aplicar las reglas de determinado filtro a un URL, a través de los elementos `<filter-mapping>` se indica que el filtro llamado `ImpuestosRegistro` sea aplicado a las solicitudes de las páginas `/javabeans/subasta.jsp` y `/java/subasta_cambio.jsp`; y,

- En el diseño del filtro descrito en la sección anterior en el que se hacía alusión a una cadena de filtros, dicha cadena o grupo quedaría conformada por el número de mapeos que correspondan a determinada ruta, esto es, si existen dos o tres elementos `<url-pattern>` que cumplan con la ruta, la cadena quedaría conformada por estos filtros, de igual manera si existiera únicamente un `<url-pattern>` que coincida con la requisición la cadena equivaldría a un solo filtro. (Latina, Menú Curso Java Web (JSP's/Servlets), 2012).

A continuación se describe la configuración del filtro en el archivo `web.xml`:

```

<!-- Archivo principal de configuracion -->
<?xml version="1.0" encoding="UTF-8"?>
<web-app id="WebApp_ID" version="2.5"
xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml
<display-name>TransWeb</display-name>
<context-param> </context-param>
<context-param> </context-param>
<context-param> </context-param>
<context-param> </context-param>
<context-param> </context-param>
<context-param> </context-param>
<context-param> </context-param>
<context-param> </context-param>
<context-param> </context-param>
<filter> </filter>
<!-- Llamada al archivo de configuracion del filtro -->
<filter>
<filter-name>AccesFilter</filter-name>
<filter-class>ec.com.entico.agencia.web.actions.AccesActionFilter</filter-class>
<init-param>
<!-- paginas que no pasan por el filtro -->
<param-name>paginasNoProtegidas</param-name>
<param-value>index.html,index.jsp,main.jsf,salir.jsf,sinPermisos.jsf,login.jsf,login-error.jsf,error404.jsf,proxyAccess.do,selMo
</init-param>
</filter>
<filter-mapping>
<filter-name>ajax4jsf</filter-name>
<servlet-name>Faces Servlet</servlet-name>
<dispatcher>REQUEST</dispatcher>
<dispatcher>FORWARD</dispatcher>
<dispatcher>INCLUDE</dispatcher>
</filter-mapping>
<!-- Paginas que pasan por el filtro -->
<filter-mapping>
<filter-name>AccesFilter</filter-name>
<url-pattern>*.jsf</url-pattern>
</filter-mapping>
<listener> </listener>
<servlet> </servlet>
<servlet>
<servlet-name>log4j-init</servlet-name>
<servlet-class>de.laliluna.logexample.Log4jConfigLoader</servlet-class>
<load-on-startup>10</load-on-startup>
</servlet>
<servlet>
<servlet-name>ProxyAccess</servlet-name>
<servlet-class>ec.com.entico.agencia.web.common.ProxyAccess</servlet-class>
</servlet>
<servlet>
<servlet-name>SeleccionModulo</servlet-name>
<servlet-class>ec.com.entico.agencia.web.common.SeleccionModulo</servlet-class>
</servlet>

```

Gráfico 61. Configuración del filtro en el archivo web.xml.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.

4.3.4.2 Configuración de archivo AccesactionFilter

```
/**
 *
 */
package ec.com.entico.agencia.web.actions;

import java.io.IOException;
import java.util.Arrays;
import java.util.StringTokenizer;

import javax.servlet.Filter;
import javax.servlet.FilterChain;
import javax.servlet.FilterConfig;
import javax.servlet.ServletException;
import javax.servlet.ServletRequest;
import javax.servlet.ServletResponse;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import org.apache.log4j.Logger;

import ec.com.entico.commons.backing.BaseBean;

/**
 * <b>Descripcion :</b><br/>
 *
 * @author jsanchez
 * @created 05/11/2009
 *
 * <b>powered by ENTICO </b>
 */
public class AccesActionFilter extends BaseBean implements Filter {

    /**
     *
     */
    private static final long serialVersionUID = 3776933014289362033L;

    private static Logger log = Logger.getLogger(AccesActionFilter.class);

    private String[] paginasNoProtegidas = null;
    StringBuffer msg = new StringBuffer();

    public void destroy() {
    }

    public void doFilter(ServletRequest servletRequest,
        ServletResponse servletResponse, FilterChain filterChain)
        throws IOException, ServletException {

        HttpServletRequest request = (HttpServletRequest) servletRequest;
        String contexto = request.getContextPath();
        String uriActual = request.getRequestURI();
        uriActual = uriActual.replace(contexto, "");
        HttpSession sesion = request.getSession();
        HttpServletResponse response = (HttpServletResponse) servletResponse;

        // si tiene componentes a4j no realiza ningun filtro
        if (uriActual.startsWith("/a4j")) {
            filterChain.doFilter(servletRequest, servletResponse);
            return;
        }
    }
}
```

Gráfico 62. Configuración de archivo AccesactionFilter; Parte a.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José.


```

        if (uriActual.startsWith("/transweb")) {
            filterChain.doFilter(servletRequest, servletResponse);
            return;
        }

        String uriSinRoot = uriActual.substring(1, uriActual.length());
        /*
         * Determina si no exiswte ninguna pagina sin proteger
         */
        if (Arrays.binarySearch(paginasNoProtegidas, uriSinRoot) >= 0) {
            filterChain.doFilter(servletRequest, servletResponse);
            return;
        }

        if (sesion.getAttribute("paUsuarioVista")==null){
            // si no se ha logueado
            msg.append("No tiene acceso al sistema");
            log.error(msg.toString());
            sesion.invalidate();
            response.sendRedirect(request.getContextPath() + "/login-error.jsf");
            return;
        }

        filterChain.doFilter(servletRequest, servletResponse);
    }

    public void init(FilterConfig filterConfig) throws ServletException {

        if (paginasNoProtegidas == null) {

            log.info("FL: 1. Carga todas las paginas no protegidas");
            paginasNoProtegidas = decodificarCadena(filterConfig
                .getInitParameter("paginasNoProtegidas"));
        }
        log.info("FL: Total de paginas no protegidas : "
            + paginasNoProtegidas.length);

        if (paginasNoProtegidas.length == 0) {
            log
                .warn("FL: No existen paginas desprotegidas del filtro principal.");
        }
        Arrays.sort(paginasNoProtegidas);
    }

    /**
     * <b>Descripcion : </b> <br/>
     * Separa las cadenas por comas y devuelve un array
     * @param cadena
     * @return
     * @author jsanchez created May 7, 2009
     */
    private String[] decodificarCadena(String cadena) {
        StringTokenizer st = new StringTokenizer(cadena, ",");
        String[] cadenaToArray = new String[st.countTokens()];
        int i = 0;

        while (st.hasMoreTokens()) {
            cadenaToArray[i++] = st.nextToken();
        }
        return cadenaToArray;
    }
}

```

Gráfico 63. Configuración de archivo AccesactionFilter; Parte b.

Fuente: Sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.

Autor: Sánchez Gallardo José

CAPÍTULO V.

IMPLEMENTACIÓN Y PRUEBAS.

5.1 Pruebas de unidad

Una prueba de unidad pretende examinar cada función en un archivo de programa simple (una clase en terminología de objetos). Las librerías de pruebas de unidad formalizan éste trabajo al proporcionar clases para pruebas.

La prueba de unidad ayuda a que el módulo se haga independiente. Esto quiere decir, que un módulo que tiene una prueba de unidad se puede examinar independientemente del resto del sistema.

Estas pruebas se las va ejecutando conforme se va realizando el programa, por ejemplo, los métodos para que los clientes accedan y ejecuten las acciones definidas están programados en los objeto Dao, donde pasan su ejecución:

```

public List<EmbarqueCliente> getAllByFechaVueloAndCliente(Date fechaVuelo,
    String idCliente) {

    log.info("DAO: getAllByFechaVueloAndCliente");
    log.info("DAO: fechaVuelo : " + fechaVuelo);
    log.info("DAO: idCliente : " + idCliente);

    parametros.clear();

    parametros.put("fecha", fechaVuelo);
    parametros.put("idCliente", idCliente);

    return getJpaTemplate().findNamedQueryAndNamedParams(
        "EmbarqueCliente.findByCodigoClienteAndFecha", parametros);
}

/* (non-Javadoc)
 * @see ec.com.entico.agencia.dao.EmbarqueClienteDao#getAllByRangoFechasAndCliente(java.util.Date, java
 */
@SuppressWarnings("unchecked")
public List<EmbarqueCliente> getAllByRangoFechasAndCliente(
    Date fechaInicial, Date fechaFinal, String idCliente) {

    log.info("DAO: getAllByRangoFechasAndCliente");
    log.info("DAO: fechaInicial : " + fechaInicial);
    log.info("DAO: fechaFinal : " + fechaFinal);
    log.info("DAO: idCliente : " + idCliente);

    parametros.clear();

    parametros.put("fechaInicial", fechaInicial);
    parametros.put("fechaFinal", fechaFinal);
    parametros.put("idCliente", idCliente);

    return getJpaTemplate().findNamedQueryAndNamedParams(
        "EmbarqueCliente.findByCodigoClienteRangoFecha", parametros);
}

```

Gráfico 64. Pruebas de Unidad.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

5.2 Pruebas de funcionalidad

Una prueba de funcionalidad verifica que cada proceso realizado en el sistema sea el correcto, y que cada línea de programación sea funcional.

Las pruebas son efectuadas con cada acción que el usuario realice, por ejemplo, la funcionalidad de consulta de embarques para los clientes:

Gráfico 65. a) Pruebas de Funcionalidad.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

ACTIONS	FLIGHT DATE	AWB	CONSIGNEE	BOX BOOKED	FULL BOXES SHIPPED	PIECES SHIPPED	DEPARTURE	ARRIVAL
	12/02/2013	369-5138 8282	DESTINY FLOWERS	41.25	33.50	126	UIO 2013-02-12 13:01	MIA 2013-02-12 13:01
	12/02/2013	406-7631 3252	DESTINY FLOWERS	193.00	193.00	426	UIO 2013-02-12 12:01	MIA 2013-02-12 12:01
	12/02/2013	406-7631 3296	DESTINY FLOWERS	107.00	105.00	285	UIO 2013-02-12 12:01	MIA 2013-02-12 12:01

Gráfico 66. b) Pruebas de Funcionalidad.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

```

public class AwbReservacionMensualDaoJpa extends
    GenericJpaDao<AwbReservacionMensual, String> implements
    AwbReservacionMensualDao {

    private static final Logger log = Logger
        .getLogger(AwbReservacionMensualDaoJpa.class);

    private HashMap<String, Object> param = new HashMap<String, Object>();

    public AwbReservacionMensualDaoJpa() {
        super(AwbReservacionMensual.class);
    }

    @SuppressWarnings("unchecked")
    public List<AwbReservacionMensual> getAllByFechaVueloAndCliente(
        Date fechaVuelo, String idCliente) {

        log.info("DAO: getAllByFechaVueloAndCliente");
        log.info("DAO: fechaVuelo : " + fechaVuelo);
        log.info("DAO: idCliente : " + idCliente);

        param.clear();
        param.put("fechaVuelo", fechaVuelo);
        param.put("idCliente", idCliente);

        return getJpaTemplate().findByNameQueryAndNamedParams(
            "AwbReservacionMensual.findByFechaVuleloAndIdCliente", param);
    }
}

```

Gráfico 67. c) Pruebas de Funcionalidad.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

5.3 Pruebas de estrés

Las pruebas de estrés se aplican sobre la base de datos, enviando un número máximo de transacciones, para verificar que el sistema construido y que la base de datos no colapse.

Especificaciones del servidor:

5.3.1 Hardware

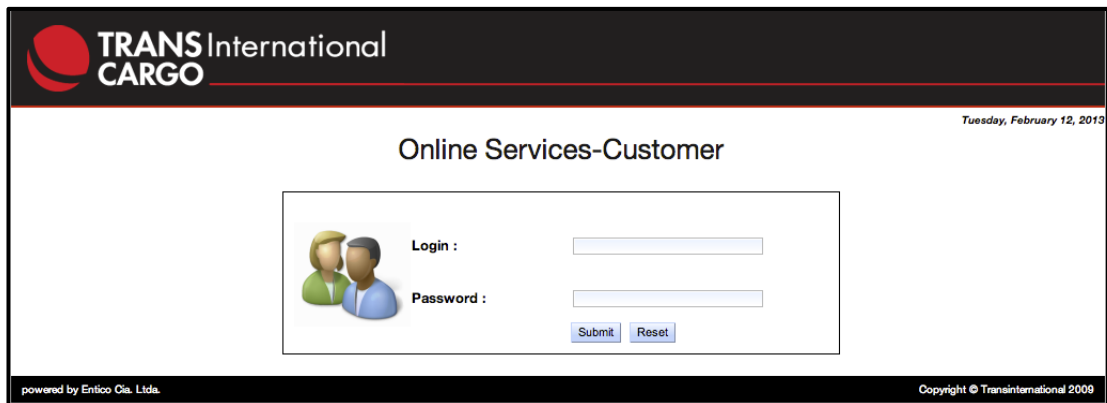
- Cpu: Clon;
- Mainboard: Interl Core i&-2600 – 3.40GHz;
- Memoria Física: 7 Gb;
- Disco Duro: 150 Gb;
- Tarjeta de red incorporada; y,
- Adicional tarjeta de red wan (10/100/1000/10000).

5.3.2 Software

- Sistema Operativo: Linux Centos;
- Versión: 2.6 32 y 64 Bits;
- Base datos: Postgres 9.1; y,
- Versión JAVA: 1.6.

5.3.3 Caso de prueba

Consulta de prueba a la tabla de reservación mensual para visualización de embarques:



The screenshot shows a web application interface for TRANS International Cargo. At the top, there is a dark header with the TRANS CARGO logo on the left and the text 'TRANS International' on the right. Below the header, the page title 'Online Services-Customer' is centered. To the right of the title, the date 'Tuesday, February 12, 2013' is displayed. The main content area contains a login form with a box for 'Login' and a box for 'Password'. Below these boxes are 'Submit' and 'Reset' buttons. The footer of the page contains the text 'powered by Entico Cia. Ltda.' on the left and 'Copyright © Transinternational 2009' on the right.

Gráfico 68. Consulta de Prueba.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

5.3.3.1 Usuario Prueba

Destiny Flowers

Usr: SOR

Pass: ****

5.3.3.2 Estado del servidor

```
top - 18:21:31 up 196 days, 42 min, 1 user, load average: 0.00, 0.06, 0.09
Tasks: 279 total, 1 running, 277 sleeping, 0 stopped, 1 zombie
Cpu0  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu1  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu2  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu3  :  2.0%us,  0.0%sy,  0.0%ni, 98.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu4  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu5  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu6  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu7  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   7975208k total,  6758892k used, 1216316k free, 126276k buffers
Swap: 10239992k total,  389680k used, 9850312k free, 2206492k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
    1 root        20   0 19348 1068  860  S   0.0   0.0   0:01.38  init
    2 root        20   0     0     0     0  S   0.0   0.0   0:00.00  kthreadd
    3 root         RT   0     0     0     0  S   0.0   0.0   0:04.58  migration/0
    4 root        20   0     0     0     0  S   0.0   0.0   2:05.19  ksoftirqd/0
    5 root         RT   0     0     0     0  S   0.0   0.0   0:00.01  migration/0
    6 root         RT   0     0     0     0  S   0.0   0.0   0:08.42  watchdog/0
    7 root         RT   0     0     0     0  S   0.0   0.0   0:06.62  migration/1
    8 root         RT   0     0     0     0  S   0.0   0.0   0:00.01  migration/1
    9 root        20   0     0     0     0  S   0.0   0.0   2:48.39  ksoftirqd/1
   10 root         RT   0     0     0     0  S   0.0   0.0   0:07.08  watchdog/1
   11 root         RT   0     0     0     0  S   0.0   0.0   0:04.54  migration/2
   12 root         RT   0     0     0     0  S   0.0   0.0   0:00.01  migration/2
   13 root        20   0     0     0     0  S   0.0   0.0   3:46.31  ksoftirqd/2
   14 root         RT   0     0     0     0  S   0.0   0.0   0:06.76  watchdog/2
   15 root         RT   0     0     0     0  S   0.0   0.0   0:04.42  migration/3
   16 root         RT   0     0     0     0  S   0.0   0.0   0:00.01  migration/3
   17 root        20   0     0     0     0  S   0.0   0.0   4:17.81  ksoftirqd/3
   18 root         RT   0     0     0     0  S   0.0   0.0   0:06.85  watchdog/3
   19 root         RT   0     0     0     0  S   0.0   0.0   0:12.88  migration/4
   20 root         RT   0     0     0     0  S   0.0   0.0   0:00.01  migration/4
   21 root        20   0     0     0     0  S   0.0   0.0   1:40.96  ksoftirqd/4
   22 root         RT   0     0     0     0  S   0.0   0.0   0:06.95  watchdog/4
   23 root         RT   0     0     0     0  S   0.0   0.0   0:27.46  migration/5
```

Gráfico 69. Estado del Servidor

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

(Observación: Ningún proceso sobre Postgres se está ejecutando en el gráfico anterior)

5.3.3.3 Procedimiento a Evaluarse

```
public class AwbReservacionMensualDaoJpa extends
    GenericJpaDao<AwbReservacionMensual, String> implements
    AwbReservacionMensualDao {

    private static final Logger log = Logger
        .getLogger(AwbReservacionMensualDaoJpa.class);

    private HashMap<String, Object> param = new HashMap<String, Object>();

    public AwbReservacionMensualDaoJpa() {
        super(AwbReservacionMensual.class);
    }

    @SuppressWarnings("unchecked")
    public List<AwbReservacionMensual> getAllByFechaVueloAndCliente(
        Date fechaVuelo, String idCliente) {

        log.info("DAO: getAllByFechaVueloAndCliente");
        log.info("DAO: fechaVuelo : " + fechaVuelo);
        log.info("DAO: idCliente : " + idCliente);

        param.clear();
        param.put("fechaVuelo", fechaVuelo);
        param.put("idCliente", idCliente);

        return getJpaTemplate().findByNameQueryAndNamedParams(
            "AwbReservacionMensual.findByFechaVueloAndIdCliente", param);
    }
}
```

Gráfico 70. Procedimiento a Evaluarse.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

5.3.3.4 Petición a la página con 100 hilos

- Resultado:

```
top - 18:44:47 up 196 days, 1:05, 1 user, load average: 0.04, 0.19, 0.11
Tasks: 278 total, 1 running, 277 sleeping, 0 stopped, 0 zombie
Cpu0  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu1  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu2  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu3  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu4  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu5  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu6  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu7  :  0.7%us,  0.7%sy,  0.0%ni, 98.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   7975208k total,  6803780k used, 1171428k free, 127804k buffers
Swap: 10239992k total,  389572k used, 9850420k free, 2231440k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
13250	root	20	0	3542m	3.2g	3460	S	1.3	41.5	18:00.30	java
1	root	20	0	19348	1068	860	S	0.0	0.0	0:01.38	init
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	RT	0	0	0	0	S	0.0	0.0	0:04.58	migration/0
4	root	20	0	0	0	0	S	0.0	0.0	2:05.21	ksoftirqd/0
5	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/0
6	root	RT	0	0	0	0	S	0.0	0.0	0:08.42	watchdog/0
7	root	RT	0	0	0	0	S	0.0	0.0	0:06.62	migration/1
8	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/1
9	root	20	0	0	0	0	S	0.0	0.0	2:48.41	ksoftirqd/1
10	root	RT	0	0	0	0	S	0.0	0.0	0:07.08	watchdog/1
11	root	RT	0	0	0	0	S	0.0	0.0	0:04.54	migration/2
12	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/2
13	root	20	0	0	0	0	S	0.0	0.0	3:46.33	ksoftirqd/2
14	root	RT	0	0	0	0	S	0.0	0.0	0:06.76	watchdog/2
15	root	RT	0	0	0	0	S	0.0	0.0	0:04.42	migration/3
16	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/3
17	root	20	0	0	0	0	S	0.0	0.0	4:17.87	ksoftirqd/3
18	root	RT	0	0	0	0	S	0.0	0.0	0:06.85	watchdog/3
19	root	RT	0	0	0	0	S	0.0	0.0	0:12.88	migration/4
20	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/4
21	root	20	0	0	0	0	S	0.0	0.0	1:40.97	ksoftirqd/4
22	root	RT	0	0	0	0	S	0.0	0.0	0:06.95	watchdog/4

Gráfico 71. Resultado de petición a la página con 100 hilos.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

5.3.3.5 Consumo de procesador cuando se ejecuta consulta sobre visualización de embarques

TRANS International
CARGO

DESTINY FLOWERS

LOGOUT

Tuesday, February 12, 2013

Main MenuPrevious ShipmentSearch by DateSearch by AWB

Online Services-Customer - Search by Date

Initial Date of Shipment : 12/02/2013Final Shipping Date : 12/02/2013

Search Shipments

ACTIONS	FLIGHT DATE	AWB	CONSIGNEE	BOX BOOKED	FULL BOXES SHIPPED	PIECES SHIPPED	DEPARTURE	ARRIVAL
	12/02/2013	369-5138 8282	DESTINY FLOWERS	41.25	33.50	126	UIO 2013-02-12 13:01	MIA 2013-02-12 11:01
	12/02/2013	406-7631 3252	DESTINY FLOWERS	193.00	193.00	426	UIO 2013-02-12 12:01	MIA 2013-02-12 11:01
	12/02/2013	406-7631 3286	DESTINY FLOWERS	107.00	105.00	285	UIO 2013-02-12 12:01	MIA 2013-02-12 11:01

Gráfico 72. a) Consumo de procesador en visualización de embarques.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José.

- Resultado en procesador:

```
top - 18:57:28 up 196 days, 1:18, 1 user, load average: 0.02, 0.04, 0.05
Tasks: 279 total, 2 running, 277 sleeping, 0 stopped, 0 zombie
Cpu0  :  0.7%us,  0.7%sy,  0.0%ni, 98.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu1  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu2  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu3  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu4  :  0.7%us,  0.7%sy,  0.0%ni, 98.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu5  :  1.3%us,  0.0%sy,  0.0%ni, 98.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu6  : 21.5%us,  0.0%sy,  0.0%ni, 78.5%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu7  :  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   7975208k total, 6816072k used, 1159136k free, 128356k buffers
Swap: 10239992k total, 389564k used, 9850428k free, 2241400k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU  %MEM    TIME+  COMMAND
 29403 postgres  20   0 228m  13m  11m R 21.9   0.2   0:00.33 postmaster
13250 root       20   0 3542m 3.2g 3460 S  0.7 41.5 18:04.25 java
   1 root       20   0 19348 1068  860 S  0.0  0.0   0:01.38 init
   2 root       20   0     0     0     0 S  0.0  0.0   0:00.00 kthreadd
   3 root       RT    0     0     0     0 S  0.0  0.0   0:04.58 migration/0
   4 root       20   0     0     0     0 S  0.0  0.0   2:05.21 ksoftirqd/0
   5 root       RT    0     0     0     0 S  0.0  0.0   0:00.01 migration/0
   6 root       RT    0     0     0     0 S  0.0  0.0   0:08.42 watchdog/0
   7 root       RT    0     0     0     0 S  0.0  0.0   0:06.62 migration/1
   8 root       RT    0     0     0     0 S  0.0  0.0   0:00.01 migration/1
   9 root       20   0     0     0     0 S  0.0  0.0   2:48.41 ksoftirqd/1
  10 root       RT    0     0     0     0 S  0.0  0.0   0:07.08 watchdog/1
  11 root       RT    0     0     0     0 S  0.0  0.0   0:04.54 migration/2
  12 root       RT    0     0     0     0 S  0.0  0.0   0:00.01 migration/2
  13 root       20   0     0     0     0 S  0.0  0.0   3:46.33 ksoftirqd/2
  14 root       RT    0     0     0     0 S  0.0  0.0   0:06.76 watchdog/2
  15 root       RT    0     0     0     0 S  0.0  0.0   0:04.42 migration/3
  16 root       RT    0     0     0     0 S  0.0  0.0   0:00.01 migration/3
  17 root       20   0     0     0     0 S  0.0  0.0   4:17.89 ksoftirqd/3
  18 root       RT    0     0     0     0 S  0.0  0.0   0:06.85 watchdog/3
  19 root       RT    0     0     0     0 S  0.0  0.0   0:12.88 migration/4
  20 root       RT    0     0     0     0 S  0.0  0.0   0:00.01 migration/4
  21 root       20   0     0     0     0 S  0.0  0.0   1:40.97 ksoftirqd/4
```

Gráfico 73. b) Resultado en el procesador sobre la visualización de embarques, 1.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

```

top - 18:58:07 up 196 days, 1:18, 1 user, load average: 0.18, 0.08, 0.07
Tasks: 280 total, 3 running, 276 sleeping, 0 stopped, 1 zombie
Cpu0 : 0.0%us, 0.0%sy, 0.0%ni,100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu1 : 0.0%us, 0.0%sy, 0.0%ni,100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu2 : 0.7%us, 0.7%sy, 0.0%ni, 98.7%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu3 : 0.0%us, 0.0%sy, 0.0%ni,100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu4 : 0.0%us, 0.0%sy, 0.0%ni,100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu5 : 99.3%us, 0.7%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu6 : 0.0%us, 0.0%sy, 0.0%ni,100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Cpu7 : 0.0%us, 0.0%sy, 0.0%ni,100.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 7975208k total, 6820540k used, 1154668k free, 128380k buffers
Swap: 10239992k total, 389524k used, 9850468k free, 2243628k cached

```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
29456	postgres	20	0	228m	14m	12m	R	99.5	0.2	0:02.40	postmaster
1379	root	20	0	0	0	0	R	0.7	0.0	11:18.13	kondemand/2
13250	root	20	0	3542m	3.2g	3488	S	0.7	41.5	18:06.83	java
1	root	20	0	19348	1068	860	S	0.0	0.0	0:01.38	init
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	RT	0	0	0	0	S	0.0	0.0	0:04.58	migration/0
4	root	20	0	0	0	0	S	0.0	0.0	2:05.22	ksoftirqd/0
5	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/0
6	root	RT	0	0	0	0	S	0.0	0.0	0:08.42	watchdog/0
7	root	RT	0	0	0	0	S	0.0	0.0	0:06.62	migration/1
8	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/1
9	root	20	0	0	0	0	S	0.0	0.0	2:48.41	ksoftirqd/1
10	root	RT	0	0	0	0	S	0.0	0.0	0:07.08	watchdog/1
11	root	RT	0	0	0	0	S	0.0	0.0	0:04.54	migration/2
12	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/2
13	root	20	0	0	0	0	S	0.0	0.0	3:46.33	ksoftirqd/2
14	root	RT	0	0	0	0	S	0.0	0.0	0:06.76	watchdog/2
15	root	RT	0	0	0	0	S	0.0	0.0	0:04.42	migration/3
16	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/3
17	root	20	0	0	0	0	S	0.0	0.0	4:17.89	ksoftirqd/3
18	root	RT	0	0	0	0	S	0.0	0.0	0:06.85	watchdog/3
19	root	RT	0	0	0	0	S	0.0	0.0	0:12.88	migration/4
20	root	RT	0	0	0	0	S	0.0	0.0	0:00.01	migration/4

Gráfico 74. c) Resultado en el procesador sobre la visualización de embarques, 2.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

- Resultado en log de consulta:

```

hibernate: insert into awb_detalleexportacion_tbl (awb, awborden, cajasconfirmadas, cajasreservadas, hawb, horareservacion, nomb
re_marcaacion, numrefrendo, numerodetalle, codigofloricola, peso, piezasconfirmadas, piezasreservadas, totalingreso, variedadflor
, iddetalleexportacion) values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
hibernate: insert into awb_detalleexportacion_tbl (awb, awborden, cajasconfirmadas, cajasreservadas, hawb, horareservacion, nomb
re_marcaacion, numrefrendo, numerodetalle, codigofloricola, peso, piezasconfirmadas, piezasreservadas, totalingreso, variedadflor
, iddetalleexportacion) values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
hibernate: insert into awb_rutaawb_tbl (orgaerolinea, destoaeropuerto, orgaeropuerto, awborden, destfechallegada, orgfechasalida,
desthoralllegada, horasalidorga, numvuelo, orden, awb, idruta) values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
hibernate: insert into awb_rutaawb_tbl (orgaerolinea, destoaeropuerto, orgaeropuerto, awborden, destfechallegada, orgfechasalida,
desthoralllegada, horasalidorga, numvuelo, orden, awb, idruta) values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
hibernate: update awb_rutaawb_tbl set awb=? where idruta=?
hibernate: update awb_rutaawb_tbl set awb=? where idruta=?
19:00:51|INFO |[ClienteConsultaServicioImpl.getAllByRangoFechaVueloAndIdClienteResumen:254]: SRV: getAllByRangoFechaVueloAndIdCl
ienteResumen
19:00:51|INFO |[ClienteConsultaServicioImpl.getAllByRangoFechaVueloAndIdClienteResumen:258]: SRV: usuario : SOR
19:00:51|INFO |[ClienteConsultaServicioImpl.getAllByRangoFechaVueloAndIdClienteResumen:259]: SRV: fechaInicio : Sun Jan 01 00:00
:00 ECT 2012
19:00:51|INFO |[ClienteConsultaServicioImpl.getAllByRangoFechaVueloAndIdClienteResumen:260]: SRV: fechaFinal : Tue Feb 12 00:00:
00 ECT 2013
19:00:51|INFO |[EmbarqueClienteDaoJpa.getAllByRangoFechasAndCliente:71]: DAO: getAllByRangoFechasAndCliente
19:00:51|INFO |[EmbarqueClienteDaoJpa.getAllByRangoFechasAndCliente:72]: DAO: fechaInicial : Sun Jan 01 00:00:00 ECT 2012
19:00:51|INFO |[EmbarqueClienteDaoJpa.getAllByRangoFechasAndCliente:73]: DAO: fechaFinal : Tue Feb 12 00:00:00 ECT 2013
19:00:51|INFO |[EmbarqueClienteDaoJpa.getAllByRangoFechasAndCliente:74]: DAO: idCliente : SOR
hibernate: SELECT awb_reservacionmensual_tbl.awb, awb_reservacionmensual_tbl.fecha, awb_reservacionmensual_tbl.awborden, pa_cl
iente_tbl.nombreciente, awb_cabeceraexportacion_tbl.cajasreservadas, awb_cabeceraexportacion_tbl.cajasonhand, awb_cabeceraexp
rtacion_tbl.piezasreservadas, awb_cabeceraexportacion_tbl.piezasonhand, (SELECT awb_rutaawb_tbl.orgaeropuerto || '-' || awb_rut
aawb_tbl.orgfechasalida || ' ' || awb_rutaawb_tbl.horasalidorga FROM awb_rutaawb_tbl WHERE awb_rutaawb_tbl.awb = awb_cabeceraexportacion_tbl.awb AND
awb_rutaawb_tbl.idruta = (SELECT min(awb_rutaawb_tbl.idruta)
FROM awb_rutaawb_tbl WHERE
awb_rutaawb_tbl.destfechallegada || ' ' || awb_rutaawb_tbl.desthoralllegada || ' ' || awb_rutaawb_tbl.destfechallegada FROM awb_rutaawb_tbl
WHERE awb_rutaawb_tbl.awb = awb_cabeceraexportacion_tbl.awb AND awb_rutaawb_tbl.idruta = (SELECT MAX(awb_rutaawb_tbl.idruta) AS
field_1 FROM public.awb_rutaawb_tbl WHERE awb_rutaawb_tbl.awb = awb_cabeceraexportacion_tbl.awb)
) AS llegada FROM awb_reservacionmensual_tbl INNER JOIN awb_cabeceraexportacion_tbl ON (awb_reservacionmensual_tbl.awb = awb_cabeceraexportacion_tbl.awb)
INNER JOIN pa_cliente_tbl ON (awb_reservacionmensual_tbl.codigocliente = pa_cliente_tbl.codigocliente) where awb_reservacionmensual_tbl.fecha BETWEEN ? and ? and pa_cliente_tbl.Usuario = ? ORDER BY awb_reservacionmensual_tbl.fecha, pa_cliente_tbl.nombreciente

```

Gráfico 75. d) Resultado en el procesador sobre la visualización de embarques en long.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

- Observaciones:

Ya que el procesador cuenta con 7 núcleos, no se muestra saturación en los mismos, por lo que el resultado no produce que la base de datos se sature de transacciones o peticiones.

5.4 Manuales técnicos y para el usuario

5.4.1. Manual técnico

Por requerimientos de la empresa, el servidor no se encuentra alojado en la misma; se ha alquilado un equipo a la empresa ESECUREDATA, la misma se encarga de la instalación del sistema operativo.

5.4.1.1 Instalación de Postgres 9.1

- *Desactivar los repositorios por defecto de CentOS:*

Para instalar la versión 9.1 de PostgreSQL se deshabilitan los repositorios de PostgreSQL que vienen con el sistema operativo. Sino se hace éste paso, es posible que sólo se instale la versión que proporcionan los paquetes de la distribución en cuestión.

Para desactivar dichos repositorios se edita el fichero /etc/yum.repos.d/CentOS-Base.repo.

Y en las secciones base y updates se excluye PostgreSQL, para ello añadir: `exclude=postgresql*` al final de cada sección. Dichas secciones deben acabar pareciéndose a lo siguiente:

```
[base]
name=CentOS-$releasever - Base
mirrorlist=http://mirrorlist.centos.org/?release=$
releasever&arch=$basearch&rep$
#baseurl=http://mirror.centos.org/centos/$releasev
er/os/$basearch/
gpgcheck=1
gpgkey=http://mirror.centos.org/centos/RPM-GPG-
KEY-CentOS-5
exclude=postgresql*
[updates]
name=CentOS-$releasever - Updates
mirrorlist=http://mirrorlist.centos.org/?release=$
releasever&arch=$basearch&rep$
#baseurl=http://mirror.centos.org/centos/$releasev
er/updates/$basearch/
gpgcheck=1
gpgkey=http://mirror.centos.org/centos/RPM-GPG-
```

Gráfico 76. a) Secciones Base y Updates Modificadas.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

- *Añadir los repositorios de PostgreSQL 9.1*

Una vez desactivados los repositorios por defecto se definen los nuevos repositorios. Para ello utilizar los RPM de la página de repositorios de PostgreSQL. En la página copiar el enlace de la última versión de la rama 9.1 a la línea de comandos:

```
$> cd /tmp

$> wget http://yum.pgsqlrpms.org/reporpms/8.3/pgdg-
centos-8.3-6.noarch.rpm

$> rpm -ivh pgdg-centos-8.3-6.noarch.rpm
```

Gráfico 77. b) Enlace de última versión de la rama 9.1

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

NOTA: Cambiar la URL de descarga del paquete RPM por la que corresponda en el momento se realiza la operación.

<http://ixavi.com/blog/2010/01/07/instalar-postgresql-8-3-en-centos-desde-yum/>

Una vez instalada la base de datos, se configura el archivo `pg_hba.conf` para que pueda aceptar las conexiones desde el exterior, donde sólo se modifican las siguientes líneas:

```

# TYPE      DATABASE      USER      ADDRESS      METHOD

# "local" is for Unix domain socket connections only
local      all         all                trust
# IPv4 local connections:
host       all         all         127.0.0.1/32  trust
host       all         all         0.0.0.0/32    trust
# IPv6 local connections:
host       all         all         ::1/128       trust
# Allow replication connections from localhost, by a user with the
# replication privilege.
#local     replication  postgres    peer
#host      replication  postgres    127.0.0.1/32  ident
#host      replication  postgres    ::1/128       ident

```

Gráfico 78. c) Configuración del Archivo pg_hba.conf.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Posterior a la instalación y modificación, se accede a la base de datos desde el cliente pg_admin instalado en la máquina que accederá al server, para crear la misma y restaurarla:

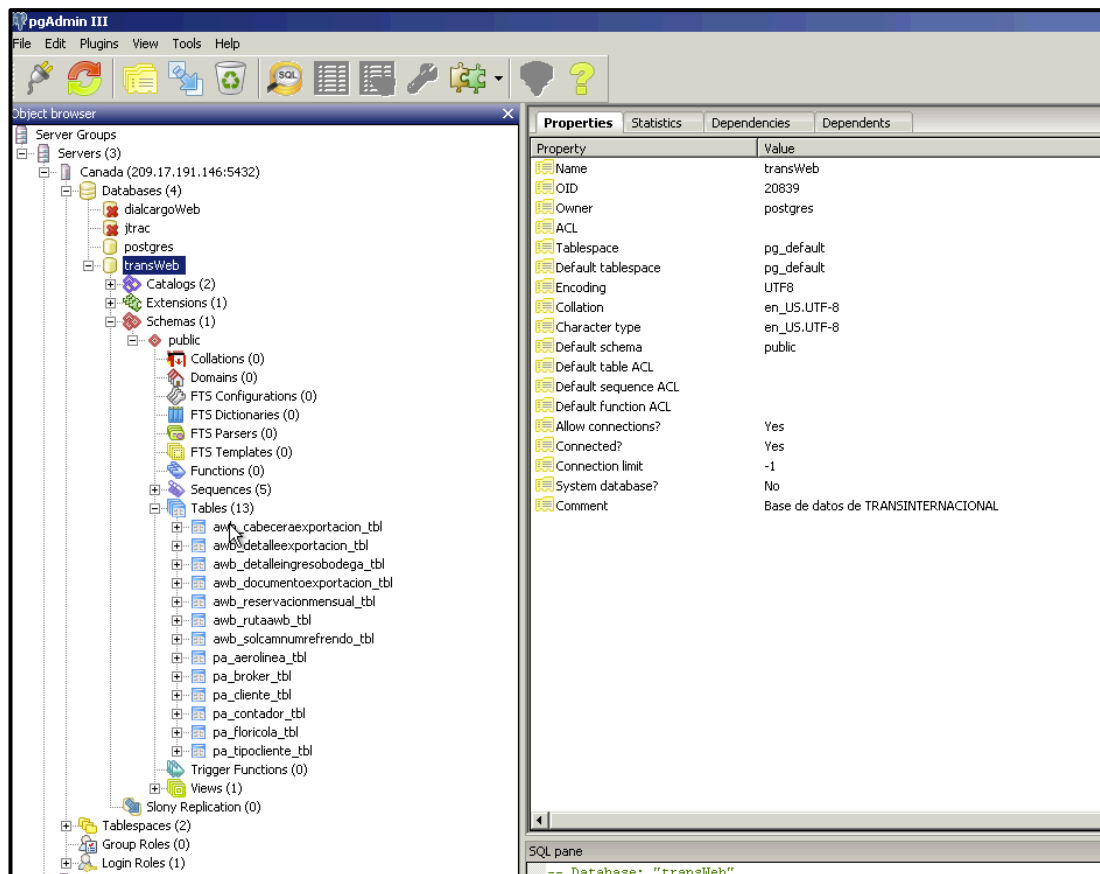


Gráfico 79. d) Entrar a la base de datos desde el cliente pg_admin instalado en la máquina.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

5.4.1.2 Instalación JAVA

“Antes de instalar JAVA, es necesario saber si existe una versión previa en el sistema GNU/Linux ejecutando la siguiente sintaxis; **rpm -qa JAVA-1.5-***; O bien; **JAVA –versión**.

Se está instalada una versión anterior a la que se desea, ésta se elimina ejecutando la siguiente sintaxis; **yum remove JAVA-1.5-***.

Se verifica si está instalado en el repositorio **rpmforge**; **yum repolist**.

```

[root@localhost ~]# yum repolist
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
* base: mirror.flhsi.com
* epel: mirror.umd.edu
* extras: mirrors.arisc.edu
* rpmforge: mirror.teklinks.com
* updates: centosy3.centos.org
repo id      repo name      status
base         CentOS-5 - Base    enabled:
cr           CentOS-5 - CR      enabled:
epel         Extra Packages for Enterprise Linux 5 - x86_64    enabled:
extras       CentOS-5 - Extras  enabled:
rpmforge     RHEL 5 - RPMforge.net - dag    enabled:
updates      CentOS-5 - Updates  enabled:
repolist: 22,486
[root@localhost ~]#

```

Gráfico 80. a) Verificación de la Instalación en el repositorio rpmforge; yum repolist.

Fuente: http://www.centosni.net/wp-content/uploads/2012/06/JAVA_1.png.

Autor: Centosni.

Para instalarlo ejecute la siguiente sintaxis; **rpm -Uvh rpmforge-release-0.5.2-2.el5.rf.x86_64.rpm**.

Y para instalar JAVA se ejecuta la siguiente sintaxis; **yum install -y JAVA-1.6.0-openjdk.x86_64**.

Package	Arch	Version	Repository	Size
Installing:				
java-1.6.0-openjdk	x86_64	1:1.6.0.0-1.25.1.10.6.el5_8	updates	36 M
Installing for dependencies:				
giflib	x86_64	4.1.3-7.3.3.el5	base	39 k
jpackage-utils	noarch	1.7.3-1jpp.2.el5	base	61 k
tzdata-java	x86_64	2012c-1.el5	updates	181 k
Transaction Summary				
Install	4 Package(s)			
Upgrade	0 Package(s)			
Total download size: 37 M				
Downloading Packages:				
(1/4): giflib-4.1.3-7.3.3.el5.x86_64.rpm			39 kB	00:02

Gráfico 81. b) Ejecución de Sintaxis.

Fuente: http://www.centosni.net/wp-content/uploads/2012/06/JAVA_21.png

Autor: Centosni.

Solo resta verificar si esta instalado ejecutando la siguiente sintaxis; **JAVA – versión”**

```
[root@localhost ~]# java -version
java version "1.6.0_22"
OpenJDK Runtime Environment (IcedTea6 1.10.6) (rhel-1.25.1.10.6.el5_8-x86_64)
OpenJDK 64-Bit Server VM (build 20.0-b11, mixed mode)
[root@localhost ~]#
```

Gráfico 82. a) Verificación de Instalación JAVA-version.

Fuente: http://www.centosni.net/wp-content/uploads/2012/06/JAVA_4.png

Autor: Centosni.

En caso de que el servicio de Postgres se detenga se procede a usar los siguientes comandos:

```
[root@mail init.d]# service postgresql-9.1 stop; service postgresql-9.1 start;
Stopping postgresql-9.1 service: [ OK ]
Starting postgresql-9.1 service: [ OK ]
[root@mail init.d]#
```

Gráfico 83. Utilización de comandos en caso de caída de Postgres.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

5.4.2. Manual de usuario

Los usuarios, Clientes, Broker y Florícolas, podrán acceder a la información de la web a través de cualquier explorador que tenga acceso a Internet; para lo cual deberá digitar la siguiente dirección:

www.ticargotracking.com

Esta dirección, mostrará la página principal de la agencia de carga TransInternacional:

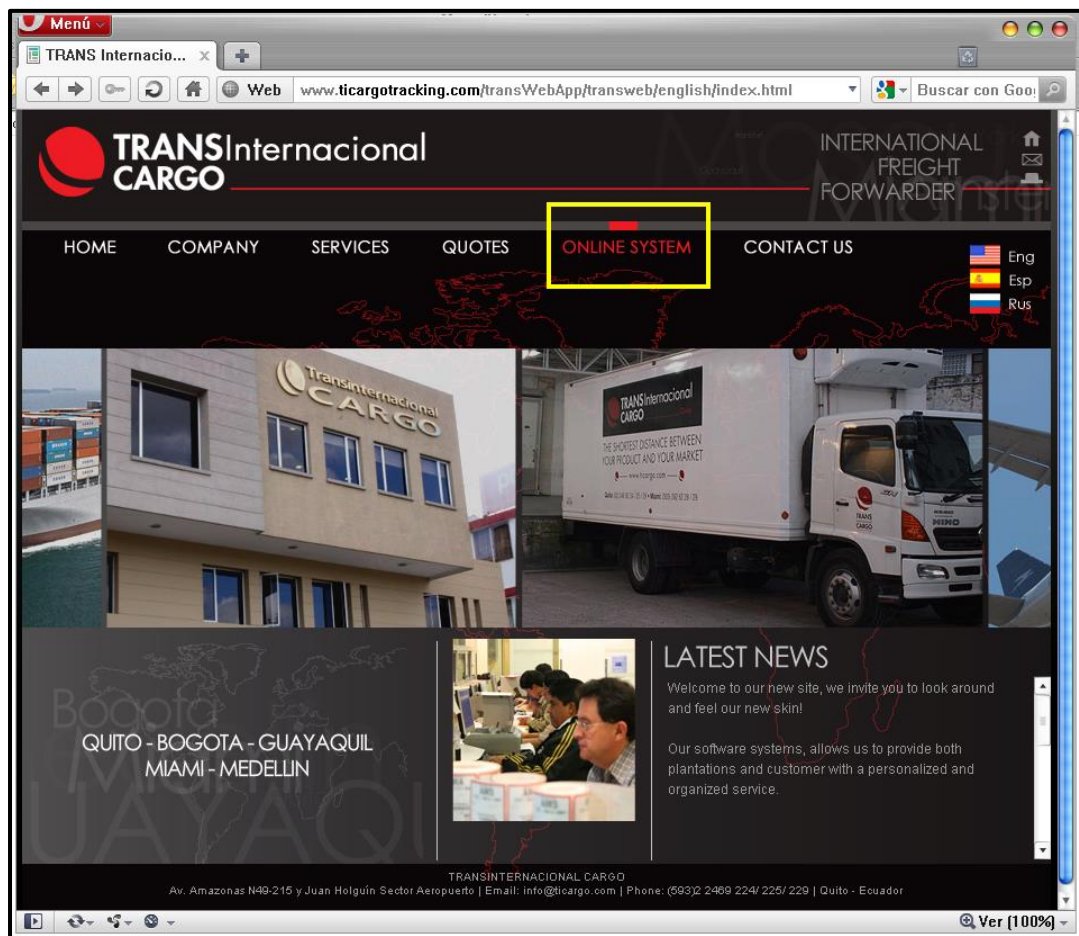


Gráfico 84. a) Página Principal Agencia de Carga TransInternacional.

Fuente: TRANS Internacional Cargo;

Autor: TRANS Internacional Cargo

Esta página mostrará un link *ONLINE SYSTEM* que mostrará la página de consulta de datos para los diferentes usuarios:

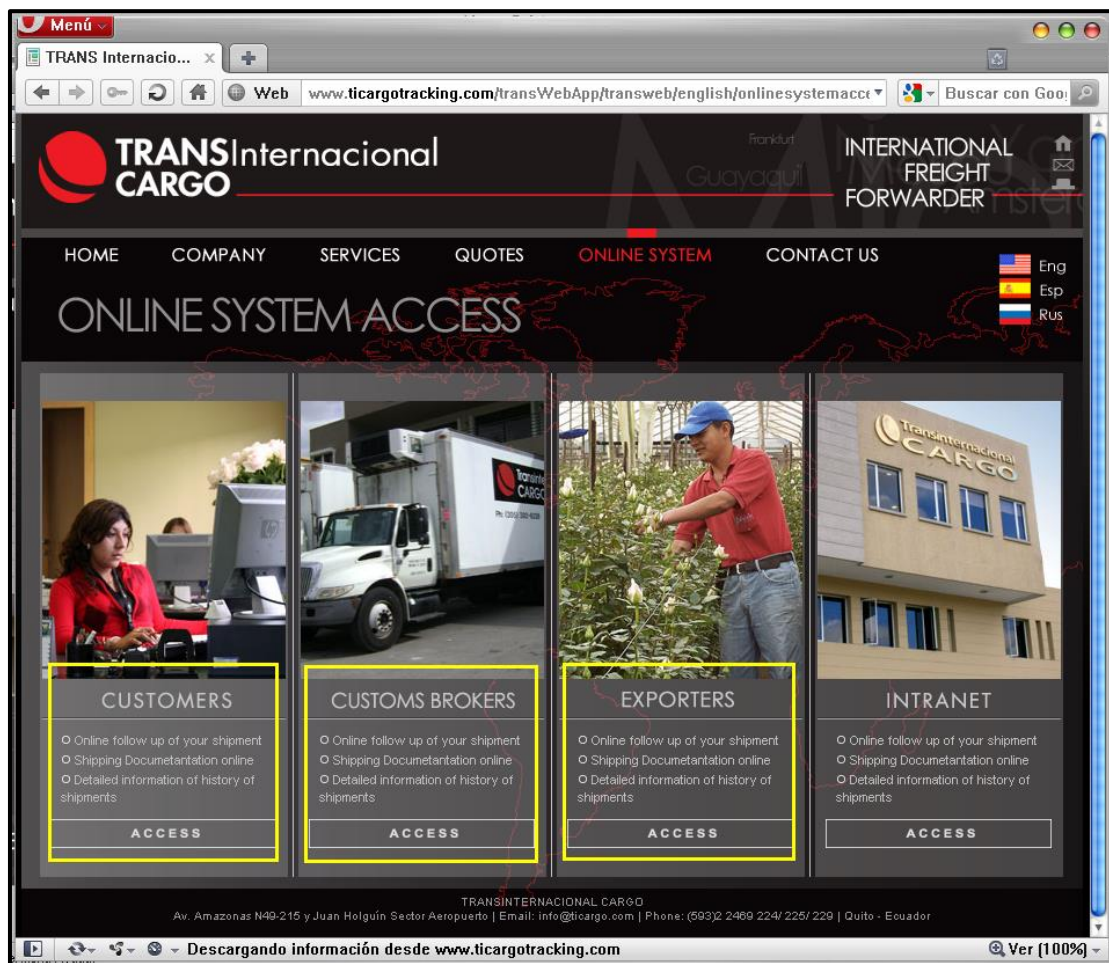


Gráfico 85. b) Página de Consulta de Datos de los Usuarios.

Fuente: TRANS Internacional Cargo;

Autor: TRANS Internacional Cargo

- *Ingreso para los Clientes:*

El link *CUSTOMERS* es el acceso específico para los clientes, en el que los usuarios podrán encontrar la información referente a las guías que le pertenecen; al dar clic en el enlace, éste le llevará a otra página donde el usuario debe autenticar sus datos para ingresar al sistema Web:



Gráfico 86. c) Acceso Específico de Clientes – link Customers.

Fuente: TRANS Internacional Cargo.

Autor: TRANS Internacional Cargo.

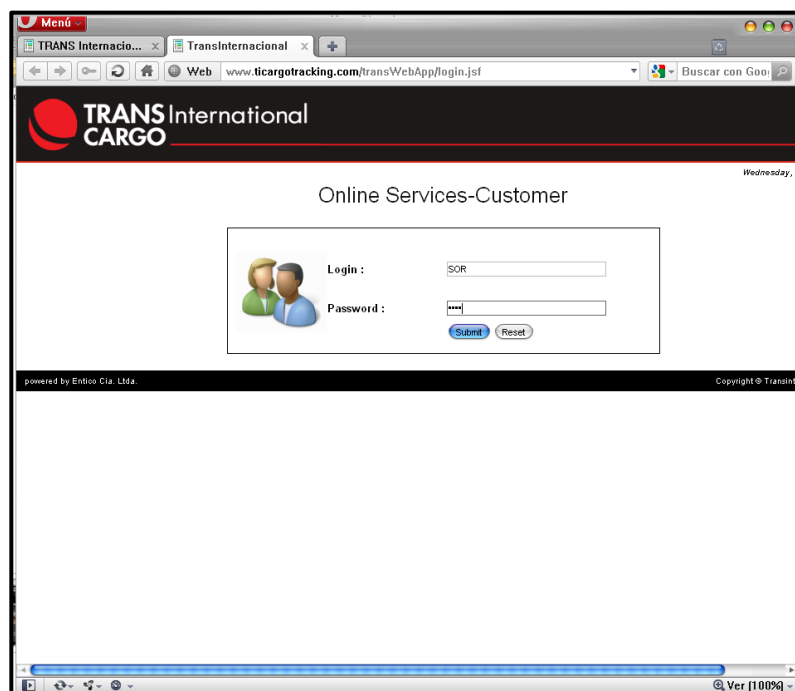


Gráfico 87. d) Visualización Acceso de Clientes.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Para que el usuario pueda autenticar sus datos, deberá solicitar a la agencia de carga su correspondiente *login* y *password*.

Luego que el usuario ingrese, encontrará 3 opciones de consultas:

- 1.- Consulta por Último Embarque (Previous Shipment);
- 2.- Consulta por Rango de Fechas (Search by Date); y,
- 3.- Consulta por Awb (Search by AWB).

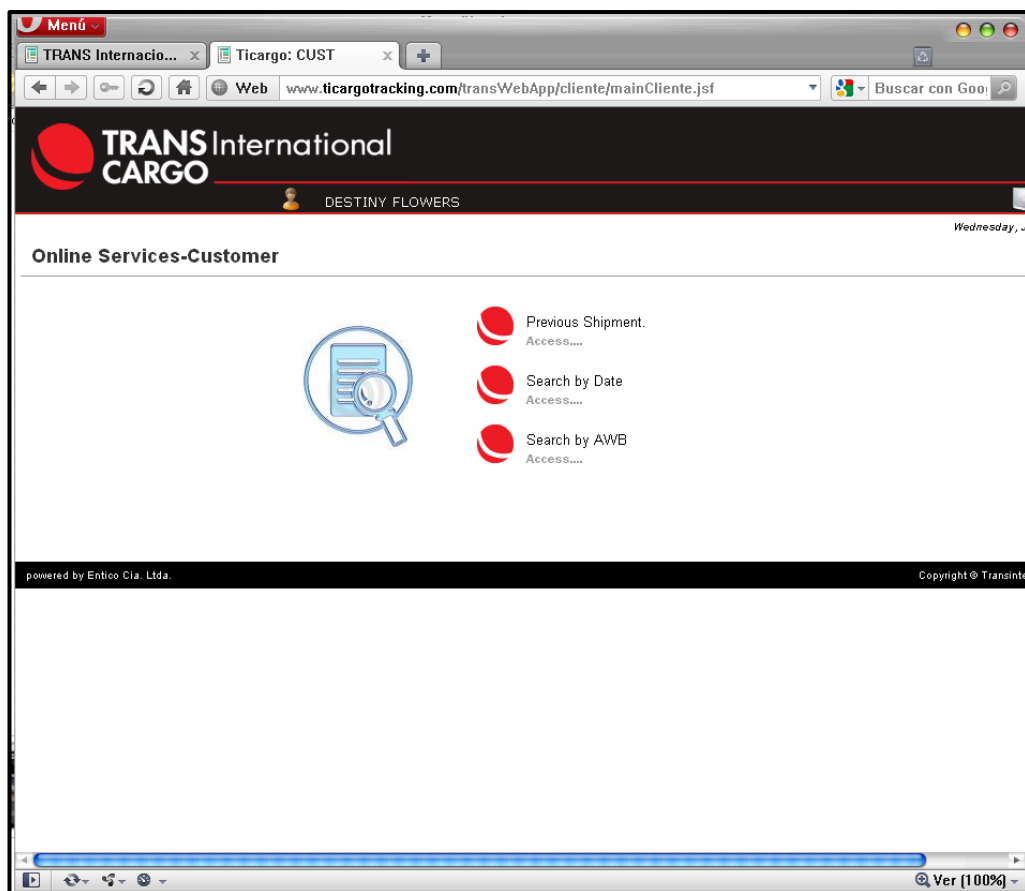


Gráfico 88. e) Visualización de las 3 opciones de Consulta.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

La primera opción *Previous Shipment* permite al usuario visualizar su último embarque:

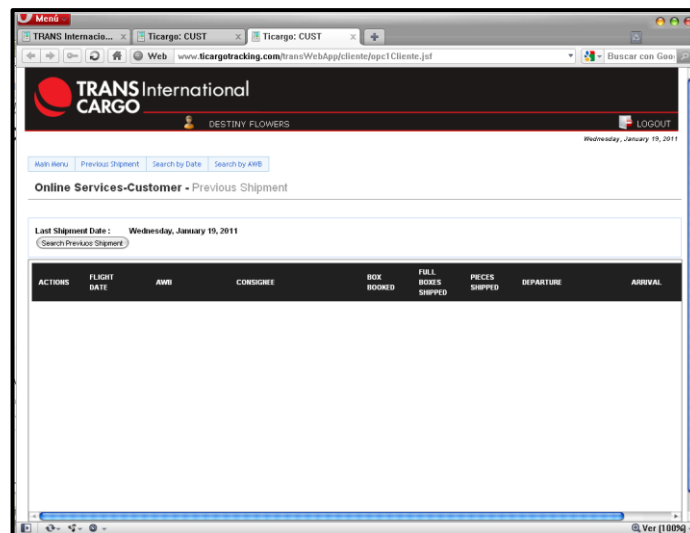


Gráfico 89. f) Previous Shipmet – Ultimo Embarque.
Fuente: TRANS Internacional Cargo.
Autor: Sánchez Gallardo José.

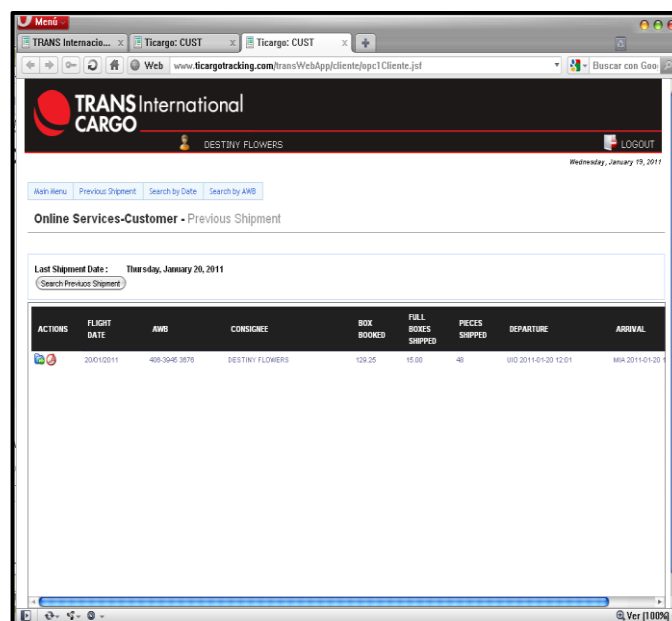


Gráfico 90. g) Previous Shipmet – Datos Principales del usuario.
Fuente: TRANS Internacional Cargo
Autor: Sánchez Gallardo José

Aquí se mostrará la última guía del cliente, con sus datos principales; si el usuario desea ver el contenido de una guía, deberá dar clic sobre la imagen que es una carpeta para visualizar los datos, o sobre la imagen con el logo de adobe Reader para descargar los documentos que pertenecen al cliente.

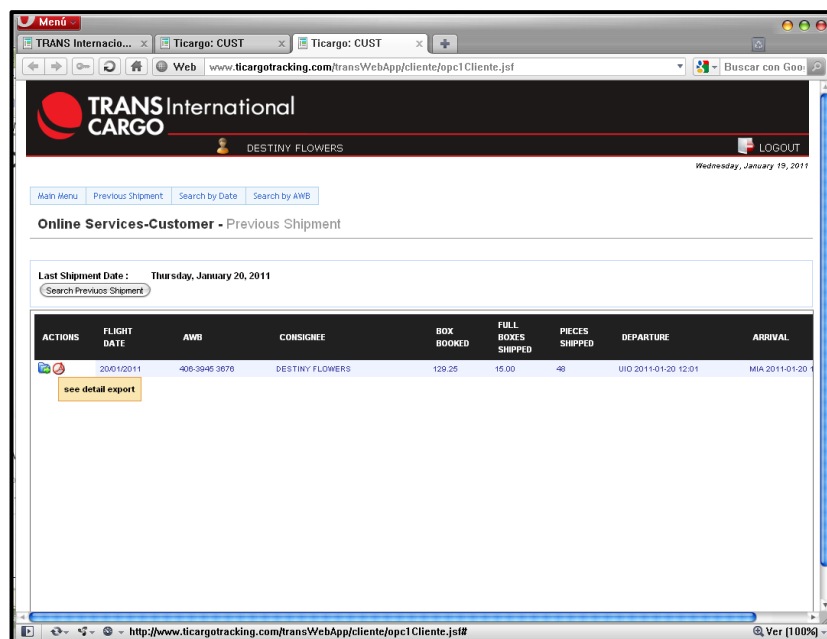


Gráfico 91. h) Previous Shipmet – Visualización de Documentos del Cliente.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

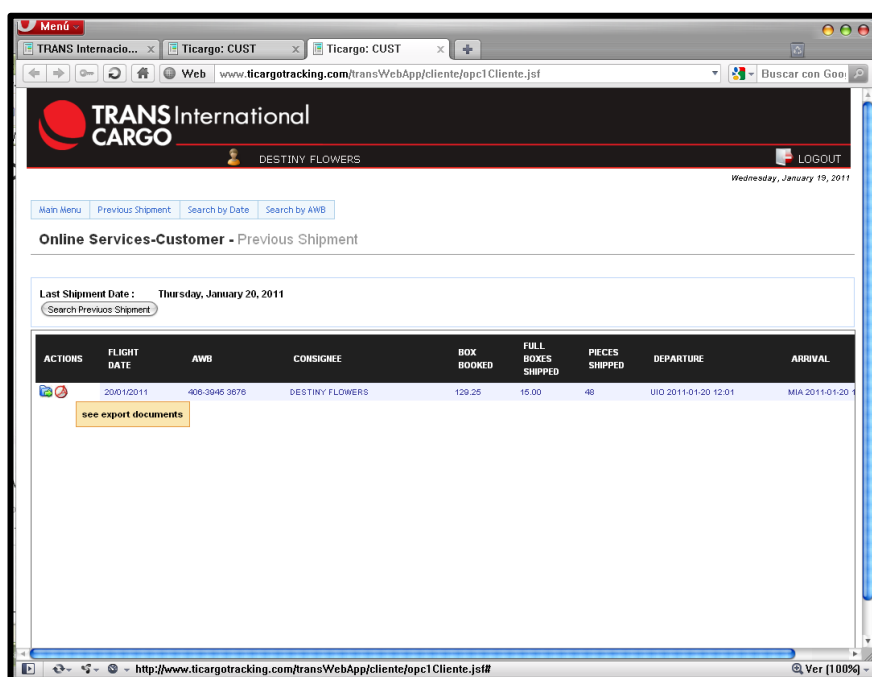


Gráfico 92. i) Previous Shipmet – Descarga de Documentos del Cliente I.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Cuando se muestra el detalle de la guía seleccionada, el cliente podrá visualizar sus datos principales, con sus respectivas florícolas coordinadas, y sus ingresos en bodega:

The screenshot shows a web browser window with multiple tabs. The active tab is titled 'TRANS Internacional...' and the address bar shows 'www.ticargotracking.com/transWebApp/cliente/detalleAwb.jsf'. The page header features the 'TRANS International CARGO' logo and the date 'Wednesday, January 19, 2011'. The main content area is titled 'Online Services-Customer - Shipping Details'.

AWB Information

AWB :	406-3945 3536
Carrier:	UNITED PARCEL SERVICE CO
Consignee:	DESTINY FLOWERS
Broker:	TRANS MIA

Flight Details

Flight No	From	Departure	To	Arrival
383	UIO	15/01/2011 - 12:01	MIA	15/01/2011 - 19:01

Exporter Information
AWB Distribution : 123-0620 4539
Client Distribution : ROSAP CLAPPERS WHOLESALE

Exporter	HAWB	Full Boxes Booked	Full Boxes Shipped	Full Boxes Received	Pieces Received	Delivery date and Time	Temp C	Observations
ROSAPRIMA CIA.LTDA.	10170317	0.50	0.50	0.50	1	14/01/2011 - 19:47:57	6.5	
		0.50	0.50	0.50	1			

Exporter Information
AWB Distribution : 123-0620 4507
Client Distribution : BELLAFLOR JL FRESH WHOLE

Gráfico 93. j) Previous Shipmet – Visualización de datos principales del Cliente.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José

Cuando el cliente desee descargar los documentos, se le mostrará los tipos de documentos que tiene para descargar o visualizar, dependiendo del browser que esté usando:

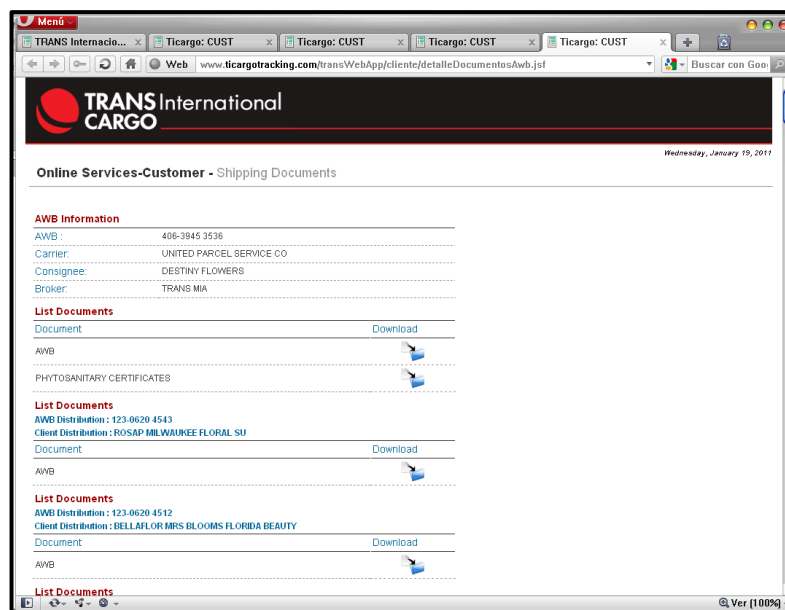


Gráfico 94. k) Previous Shipmet – Opciones de Descarga de Documentos del Cliente.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Seleccionando cualquier tipo de documento, podrá descargarlo o visualizarlo:

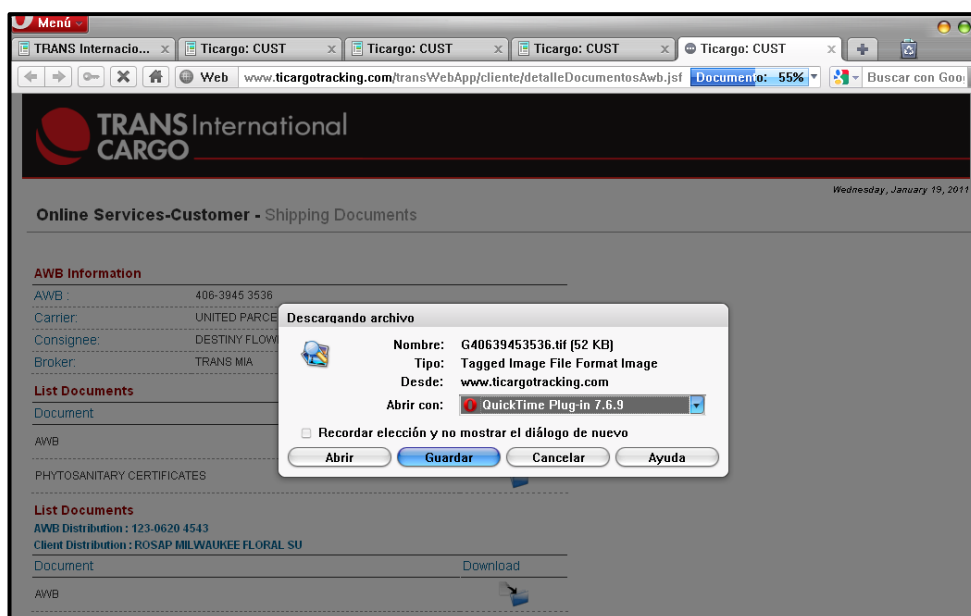


Gráfico 95. l) Previous Shipmet –Descarga de Documentos del Cliente II.

Fuente: TRANS Internacional Cargo

Autor: Sánchez Gallardo José.



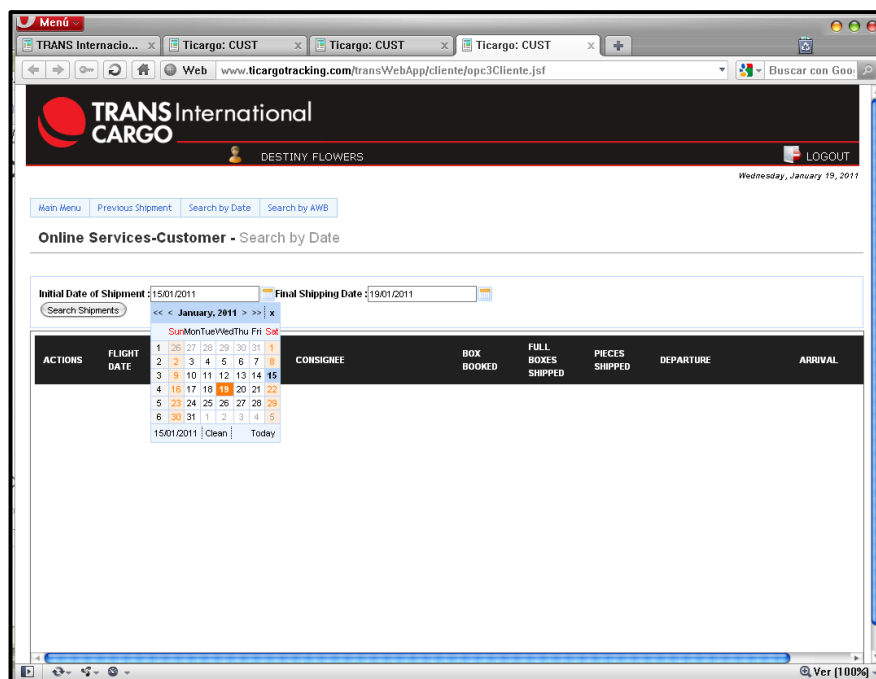


Gráfico 96. n) Search by Date – Selecccion de Rango de fechas 1.
Fuente: TRANS Internacional Cargo
Autor: Sánchez Gallardo José

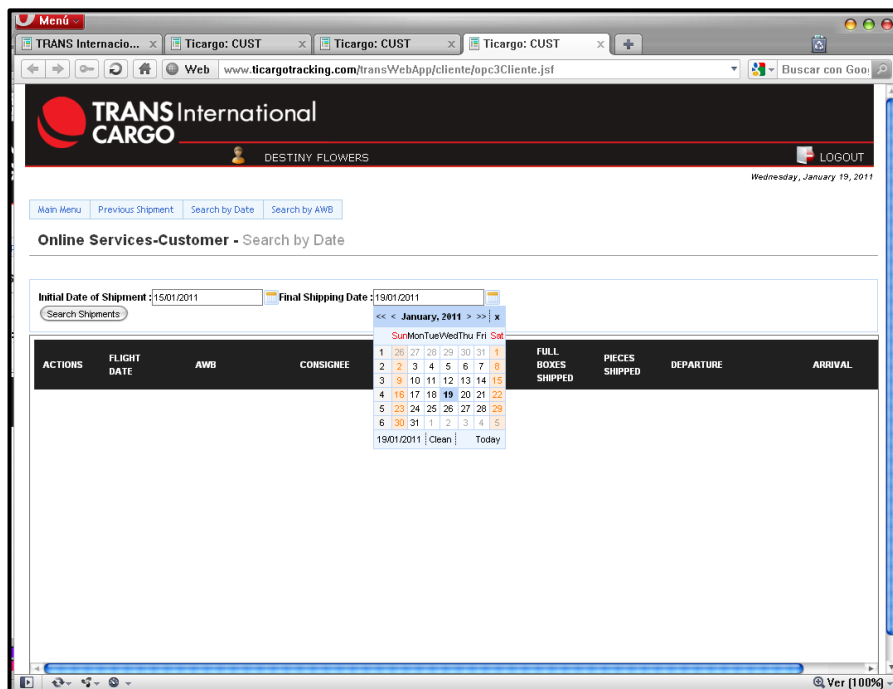


Gráfico 97. o) Search by Date – Selecccion de Rango de fechas 2.
Fuente: TRANS Internacional Cargo
Autor: Sánchez Gallardo José

Seleccionando el rango de fechas, el sistema mostrará los datos que se encuentren entre dichas fechas:

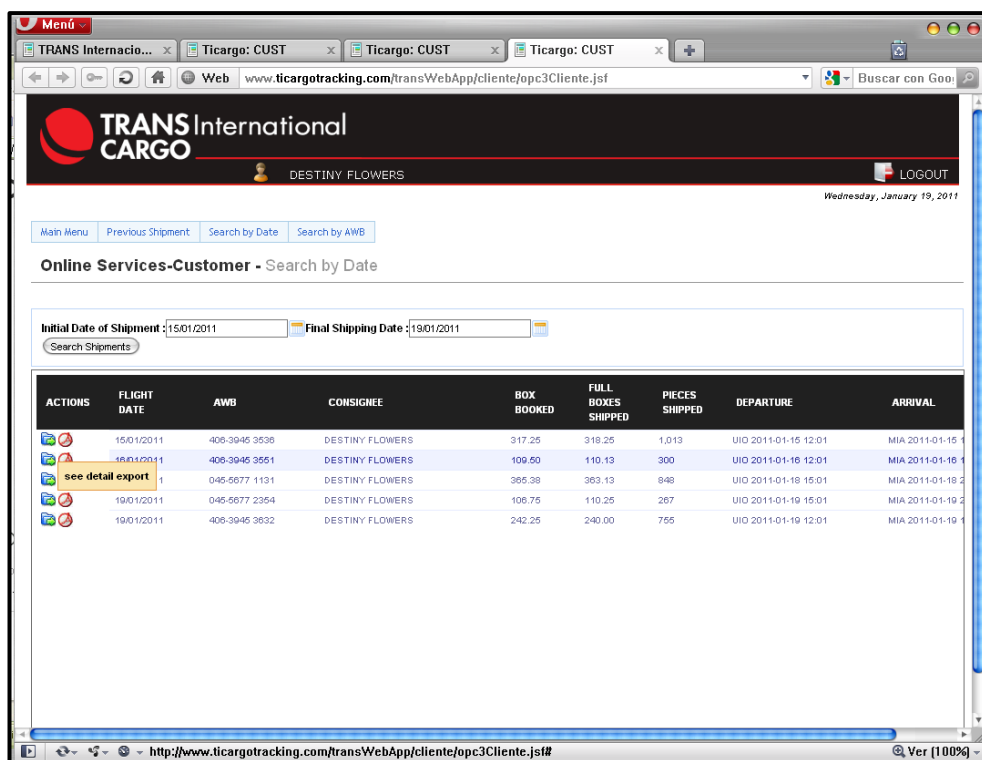


Gráfico 98. p) Search by Date – Parámetro de fechas 3.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Posteriormente, el proceso para poder visualizar las guías será el mismo que se describió anteriormente.

La última opción Search by AWB permite al cliente buscar una guía específica mediante su Número:

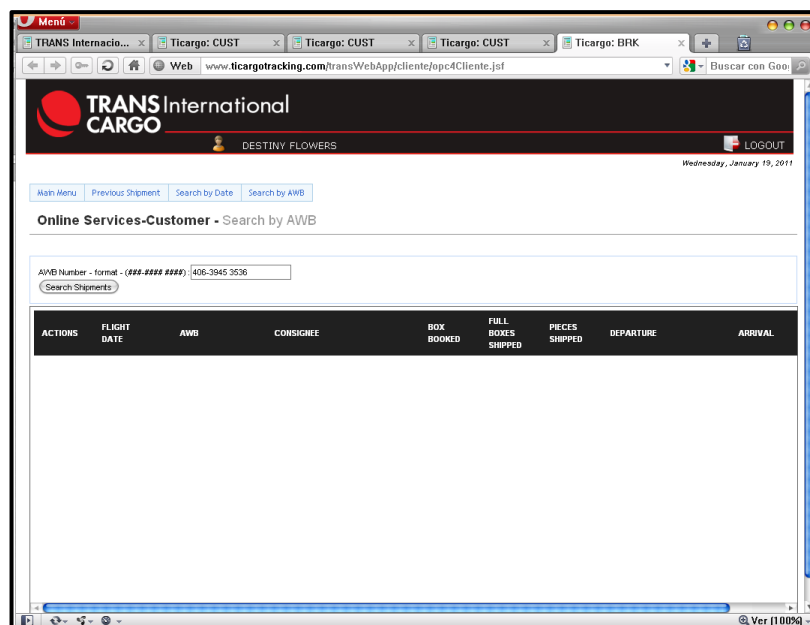


Gráfico 99. q) Search by AWB.
Fuente: TRANS Internacional Cargo.
Autor: Sánchez Gallardo José.

Si se encuentra información de acuerdo al número, se mostrará la guía consultada:

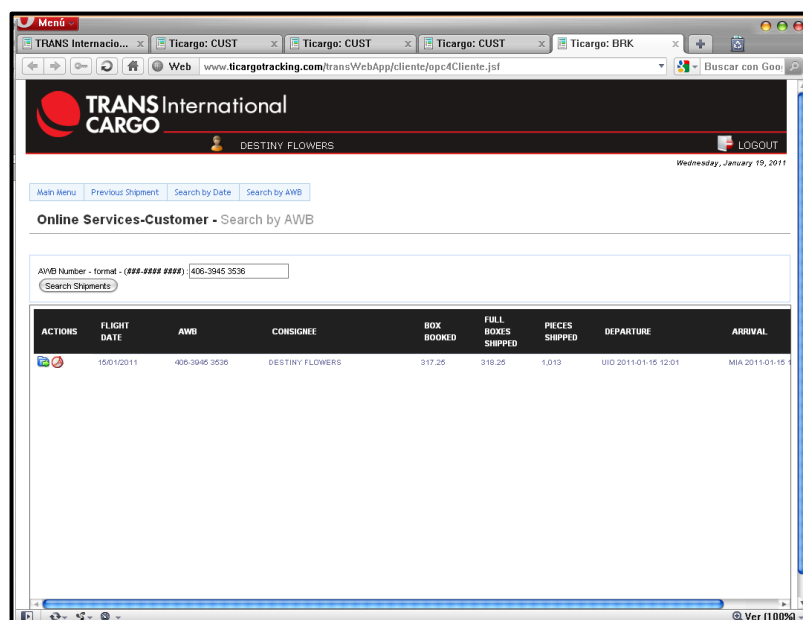


Gráfico 100. r) Search by AWB – Guía consultada de Acuerdo al número.
Fuente: TRANS Internacional Cargo.
Autor: Sánchez Gallardo José.

Posteriormente, el proceso para poder visualizar las guías será el mismo que se describió anteriormente.

- *Ingreso para los Broker:*

El link *CUSTOMS BROKERS* es el acceso específico para los brókers, en el que los usuarios podrán encontrar la información referente a las guías que les corresponde; al dar click en el enlace, éste le llevará a otra página donde el usuario debe autenticar sus datos para ingresar al sistema Web:

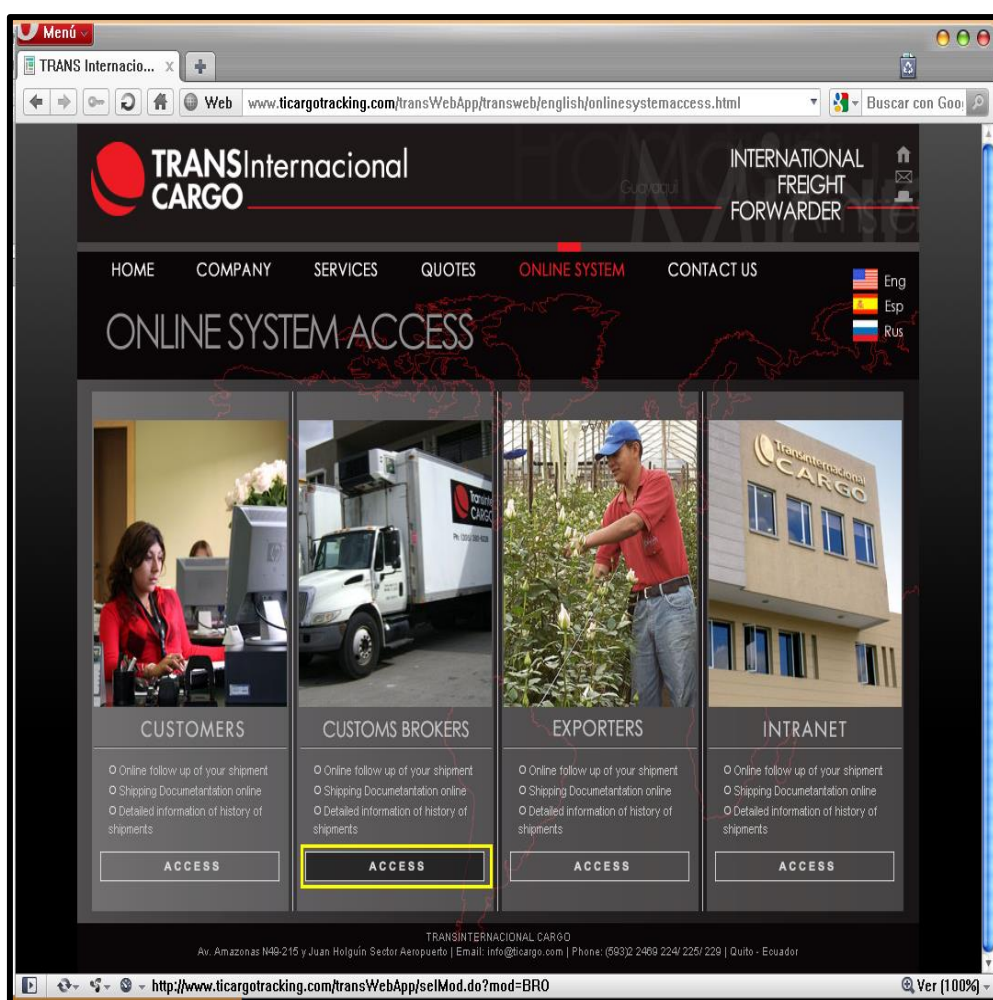


Gráfico 101. s) link CUSTOMS BROKERS.

Fuente: TRANS Internacional Cargo.

Autor: TRANS Internacional Cargo.

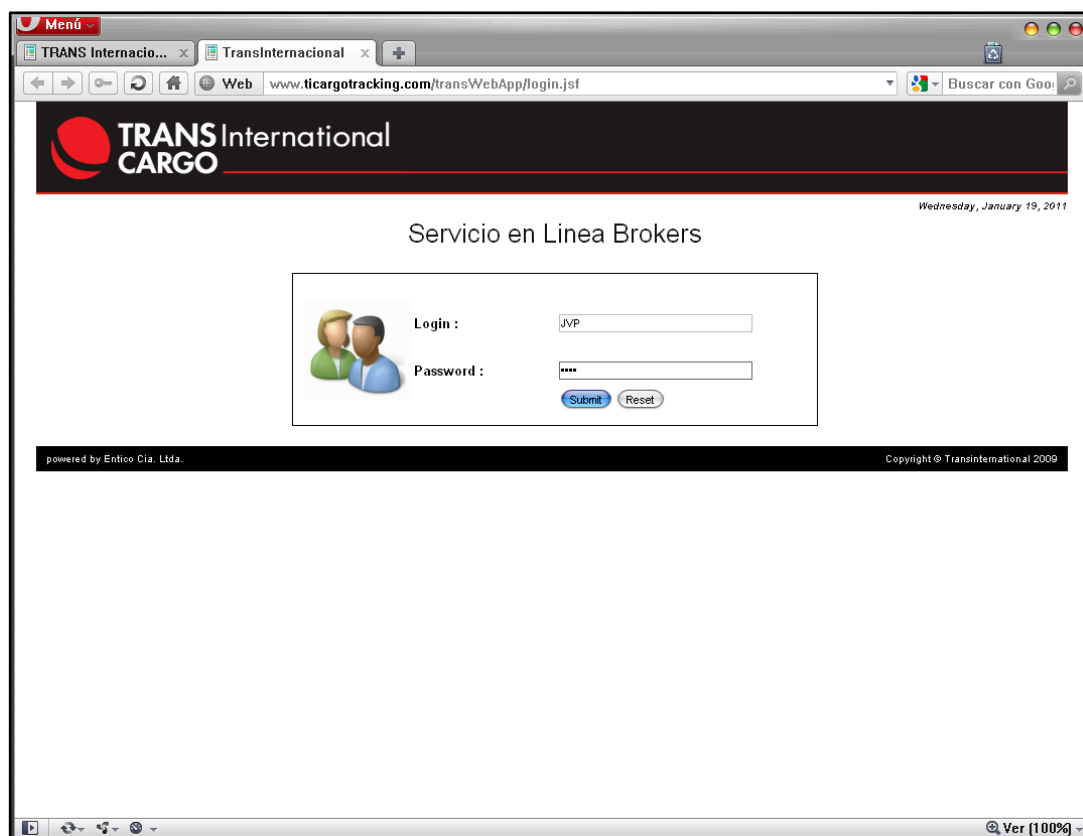


Gráfico 102. t) link CUSTOMS BROKERS – Visualización de Acceso.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Para que el usuario pueda autenticar sus datos, deberá solicitar a la agencia de carga su correspondiente *login* y *password*.

Luego de que el usuario ingrese, encontrará 4 opciones de consultas:

- 1.- Consulta por Último Embarque;
- 2.- Consulta por Búsqueda de Awb por fecha de vuelo;
- 3.- Consulta por Awb por Rango de Fechas; y,
- 4.- Consulta por Búsqueda por número de Awb específico.

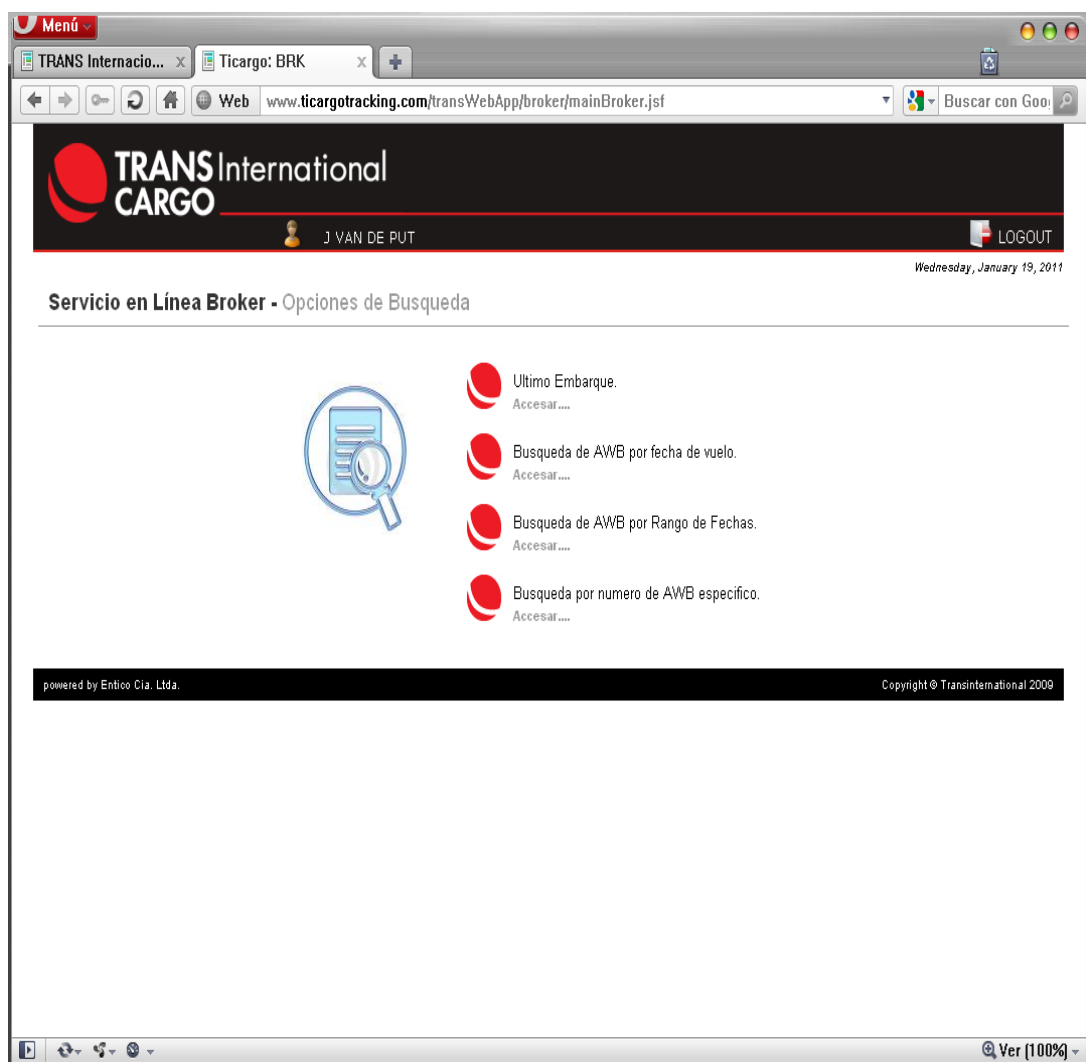


Gráfico 103. u) Visualización de las 4 opciones de Consulta.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

La primera opción *Consulta por Último Embarque*, mostrará la información de la última guía del usuario:

ACTIONS	FECHA VUELO	AWB	CUSTOMER	MASTER AWB	MASTER CUSTOMER	BOX BOOKED	BOX SHIPPED	PCS SHIPPED	ETD	ETA
2	20/01/2011	129-5903 5950	ALMEI	-	-	12.00	8.50	17	UID 20/01/2011 17:01	SPL 20/
2	20/01/2011	074-4315 5952	BLUMEN GALLI AG	-	-	21.25	0.00	0	UID 20/01/2011 05:01	SPL 20/
2	20/01/2011	129-5903 5945	CAMEL DE NATALIA DE VIELEZ	-	-	10.50	0.00	0	UID 20/01/2011 15:01	SPL 20/
2	20/01/2011	129-5903 5953	MARK FLOWERS AZALIA GROUP	-	-	12.00	1.50	3	UID 20/01/2011 17:01	SPL 20/
2	20/01/2011	129-5903 5724	MARK FLOWERS OSTAP GLADIATOR	-	-	7.50	7.50	15	UID 20/01/2011 17:01	SPL 20/
2	20/01/2011	129-5903 5975	MARK FLOWERS PALMIRA GROUP	-	-	28.00	24.00	48	UID 20/01/2011 11:01	SPL 20/
2	20/01/2011	129-5903 5993	MARK FLOWERS ROEKA	-	-	9.50	0.00	0	UID 20/01/2011 17:01	SPL 20/
2	20/01/2011	129-5903 5986	UKRONIKS	-	-	20.50	14.00	28	UID 20/01/2011 23:01	SPL 20/

Gráfico 104. v) Consulta por Ultimo Embarque.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José..

En el caso de los brókers, se mostrará la información importante de la guía, y de la misma manera que para los clientes, solo se podrá descargar los documentos que tenga asignado el bróker:

AWB Information	
AWB :	074-4315 5936
Carrier:	KLM CARGO
Consignee:	FLOWERS EXPRESS ITALY
Broker:	J VAN DE PUT

List Documents	
Document	Download
PHYTOSANITARY CERTIFICATES	
CERTIFICATE OF ORIGIN	
AWB	

Gráfico 105. w) Consulta por Ultimo Embarque - Descarga de Documentos.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

[illegible]

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

148

ACTIONS	FECHA VUELO	AWB	CUSTOMER	MASTER AWB	MASTER CUSTOMER	BOX BOOKED	BOX SHIPPED	PCS SHIPPED	ETD	ET
	11/01/2011	045-5677 1492	7 FLOWERS GROUP	-	-	195.00	198.50	423	UIO 12/01/2011 21:01	AM 15:13
	11/01/2011	020-6625 6120	7 FLOWERS TERRA LTD	-	-	103.00	103.00	241	UIO 11/01/2011 08:01	AM 17:31
	11/01/2011	045-5677 2870	AUTOMAX 2000 LTD A AMS	-	-	12.50	12.50	25	UIO 12/01/2011 21:01	AM 15:13
	11/01/2011	129-5993 5481	BLUMEN GALLI AG.	-	-	14.50	14.00	32	UIO 11/01/2011 23:01	SP 17:31
	11/01/2011	074-4315 3913	BLUMEN GALLI AG.	-	-	55.00	52.75	118	UIO 11/01/2011 05:01	SP 17:31
	11/01/2011	045-5677 3006	CRYSTALL SILVER	-	-	26.50	22.00	44	UIO 12/01/2011 21:01	AM 15:13
	11/01/2011	129-5993 5680	DELFIN 2 TATIANA DE CORDE	-	-	21.50	21.00	42	UIO 11/01/2011 21:01	SP 17:31
	11/01/2011	129-5993 5094	DE TULP AB	-	-	14.75	12.75	28	UIO 11/01/2011 22:01	SP 17:31
	11/01/2011	045-5677 1525	FIP ILIA	-	-	13.00	13.00	26	UIO 12/01/2011 21:01	AM 15:13
	11/01/2011	045-5677 1433	FLOWERS DIRECT LIMITED	-	-	26.50	26.50	53	UIO 12/01/2011 21:01	AM 15:13
	11/01/2011	129-5993 5396	FLOWERS EXPRESS ITALY	-	-	46.50	37.00	74	UIO 12/01/2011 17:01	SP 17:31

Gráfico 107. y) Consulta por Búsqueda de Awb por fecha de vuelo.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Posteriormente, el proceso para poder visualizar y descargar las guías será el mismo que se describió anteriormente.

La tercera opción *Consulta por Awb por Rango de Fechas*, permitirá realizar la búsqueda de las guías en un rango de fechas:

TRANS International CARGO

J VAN DE PUT

Logout

Wednesday, January 19, 2011

Menu Principal Ultimo Embarque AWB por Fecha AWB por Rango Fecha AWB específica

Servicio en Línea Broker - Búsqueda por Rango de Fechas

Initial Date of Shipment: 11/01/2011 Final Shipping Date: 19/01/2011

Search Shipments

ACTIONS	FECHA VUELO	AWB	CUSTOMER	MASTER AWB	MASTER CUSTOMER	BOX BOOKED	BOX SHIPPED	PCS SHIPPED	ETD	ET
	11/01/2011	045-5677 1492	7 FLOWERS GROUP	-	-	195.00	198.50	423	UIO 12/01/2011 21:01	AM 15:00
	11/01/2011	020-6625 6120	7 FLOWERS TERRA LTD	-	-	103.00	103.00	241	UIO 11/01/2011 08:01	AM 17:00
	11/01/2011	045-5677 2870	AUTOMAX 2000 LTD A AMS	-	-	12.50	12.50	25	UIO 12/01/2011 21:01	AM 15:00
	11/01/2011	129-5993 5481	BLUMEN GALLI AG.	-	-	14.50	14.00	32	UIO 11/01/2011 23:01	SP 15:00
	11/01/2011	074-4315 3913	BLUMEN GALLI AG.	-	-	55.00	52.75	118	UIO 11/01/2011 05:01	SP 15:00
	11/01/2011	045-5677 3008	CRYSTALL SILVER	-	-	26.50	22.00	44	UIO 12/01/2011 21:01	AM 15:00
	11/01/2011	129-5993 5680	DELFIN 2 TATIANA DE CORDE	-	-	21.50	21.00	42	UIO 11/01/2011 21:01	SP 15:00
	11/01/2011	129-5993 5094	DE TULP AB	-	-	14.75	12.75	28	UIO 11/01/2011 22:01	SP 15:00
	11/01/2011	045-5677 1525	FIP ILIA	-	-	13.00	13.00	26	UIO 12/01/2011 21:01	AM 15:00
	11/01/2011	045-5677 1433	FLOWERS DIRECT LIMITED	-	-	26.50	26.50	53	UIO 12/01/2011 21:01	AM 15:00
	11/01/2011	129-5993 5396	FLOWERS EXPRESS ITALY	-	-	46.50	37.00	74	UIO 12/01/2011 17:01	SP 15:00

Gráfico 109. z) Consulta por Awb por Rango de Fechas.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Posteriormente, el proceso para poder visualizar y descargar las guías será el mismo que se describió anteriormente.

La cuarta opción *Consulta por Búsqueda por número de Awb específico*, permitirá realizar la búsqueda de una guía específica de acuerdo a su número:

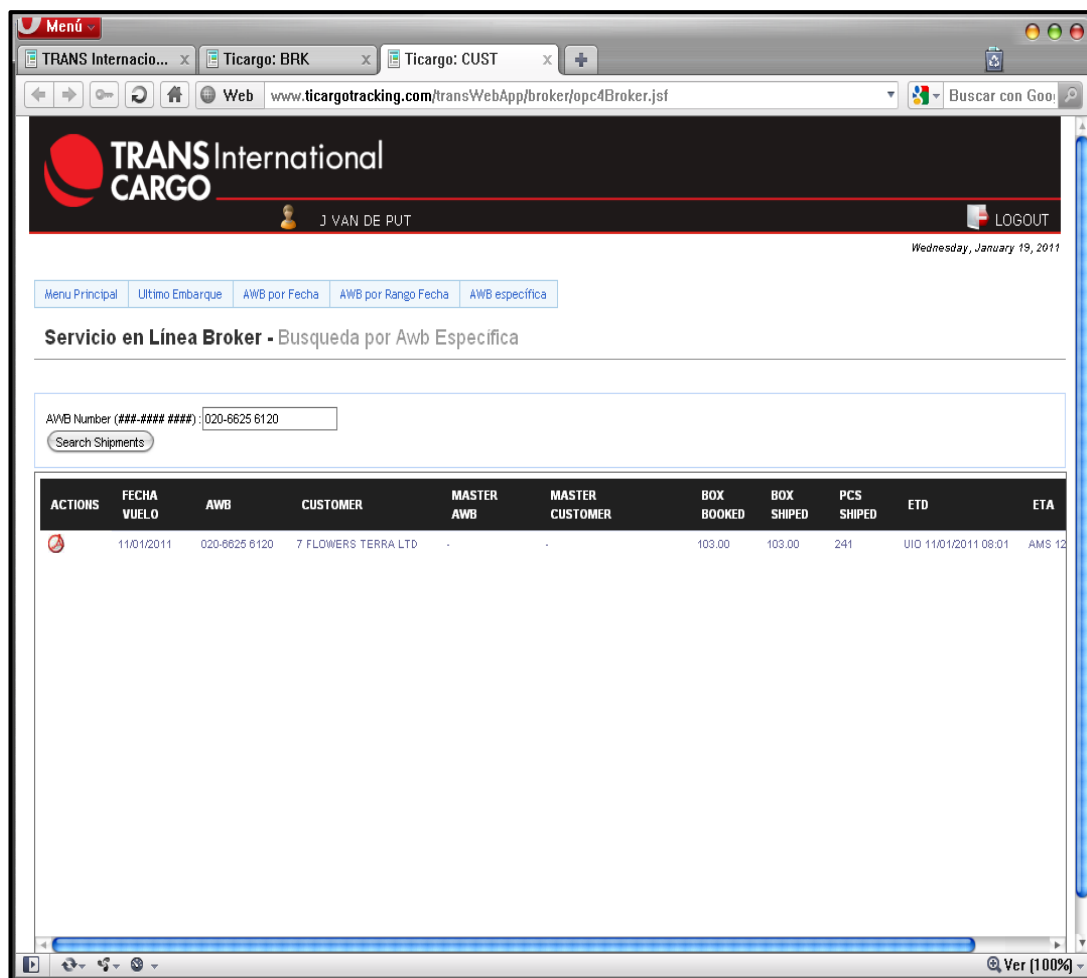


Gráfico 110. a1) Consulta por Búsqueda por número de Awb específico.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Posteriormente, el proceso para poder visualizar y descargar las guías será el mismo que se describió anteriormente.

- *Ingreso para las Florícolas:*

El link EXPORTERS es el acceso específico para las florícolas, en el que los usuarios podrán encontrar la información referente a las guías que les corresponde; al dar click en el enlace, éste le llevará a otra página donde el usuario debe autenticar sus datos para ingresar al sistema Web:

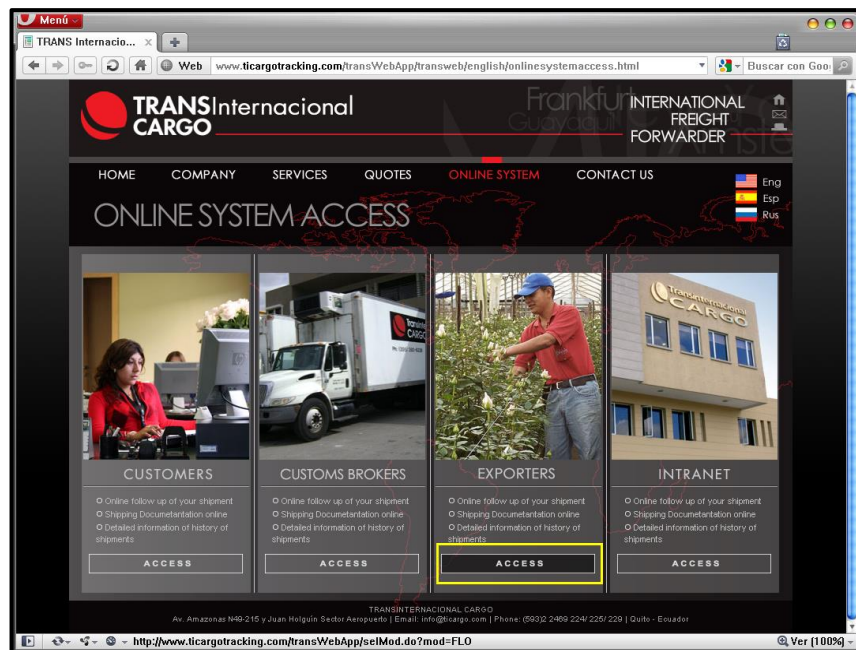


Gráfico 111. b1) link EXPORTERS - Específico para las Florícolas.

Fuente: TRANS Internacional Cargo.

Autor: TRANS Internacional Cargo.

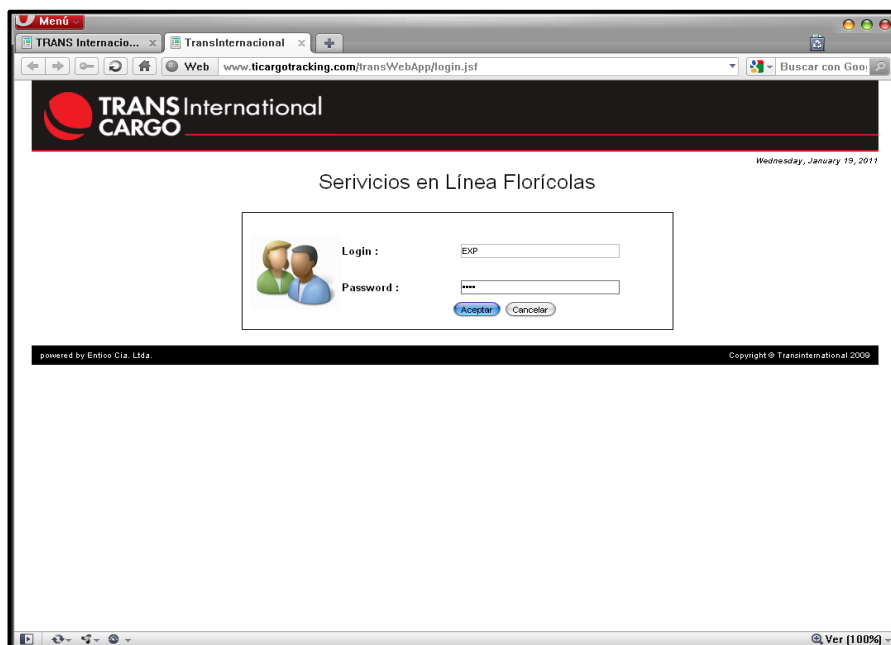


Gráfico 112. b1) Autentificación de Datos de las Florícolas.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Para que el usuario pueda autenticar sus datos, deberá solicitar a la agencia de carga su correspondiente *login* y *password*.

Luego de que el usuario ingrese, encontrará 3 opciones de consultas:

- 1.- Consulta por Búsqueda de Awb por fecha de Coordinación;
- 2.- Consulta por Awb por Rango de Fechas de Coordinación; y,
- 3.- Consulta por Búsqueda por número de Awb específico.

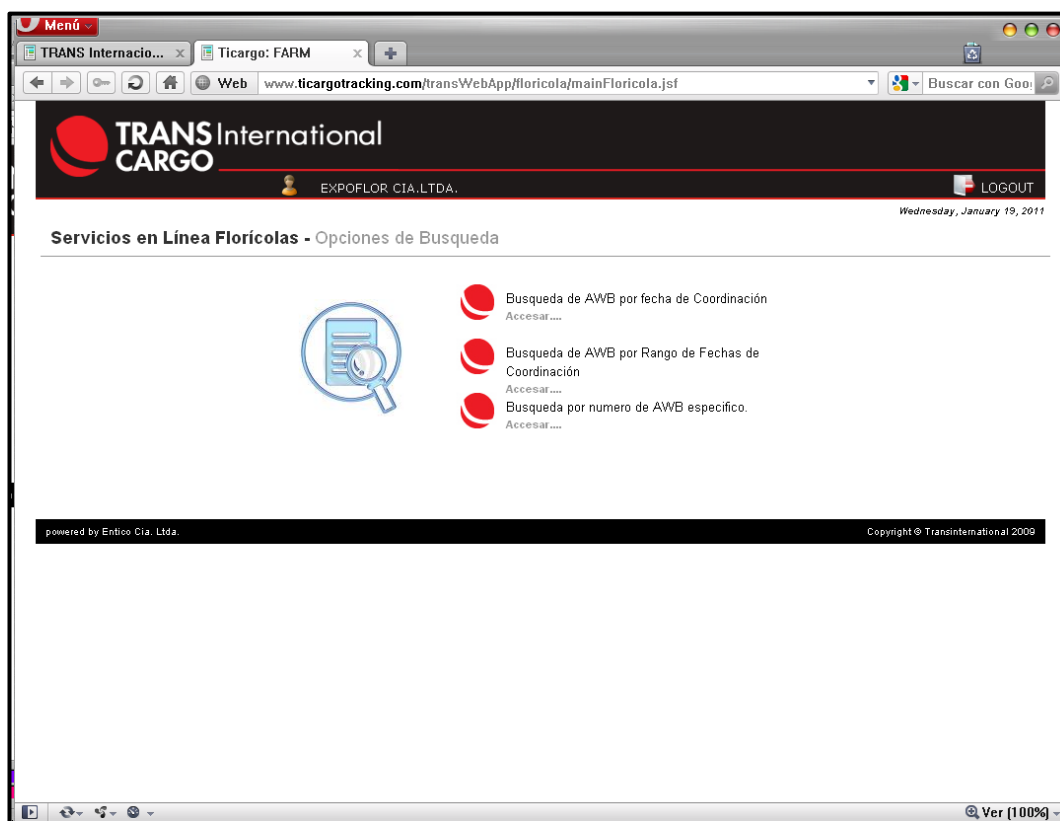


Gráfico 113. c1) EXPORTERS - Visualización de las opciones de Consulta.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

La primera opción *Consulta por Búsqueda de Awb por fecha de Coordinación*, permitirá realizar la búsqueda de las guías de la florícola en una fecha determinada:

AWB	FECHA VUELO	CLIENTE	HAWB	NUM REFRENDO	NUM MANIFESTO	BOX RSV	BOX REG	PCS REG
406-3945 3536	15/01/2011	BELLAFLOR BELLE FLORAL	10106519	055-2010-15-0594139	055-11-05-000298-8	1,500	1,250	3
406-3945 3536	15/01/2011	BELLAFLOR C AND C CUT FLO	10106525	055-2010-15-0594468	055-11-05-000298-8	1,500	1,250	3
406-3945 3536	15/01/2011	BELLAFLOR EAST COAST WHOL	10106533	055-2010-15-0594468	055-11-05-000298-8	1,000	1,000	2
406-3945 3536	15/01/2011	BELLAFLOR EMPIRE CUT FLOW	10106755	055-2010-15-0594468	055-11-05-000298-8	0,500	0,500	1
406-3945 3536	15/01/2011	BELLAFLOR FLORAL TRENDS	10106536	055-2010-15-0594468	055-11-05-000298-8	0,500	0,500	1
406-3945 3536	15/01/2011	BELLAFLOR MAHONEY'S GARDEN CENTERS	10106547	055-2010-15-0594468	055-11-05-000298-8	0,500	0,250	1
406-3945 3536	15/01/2011	BELLAFLOR P RAVENSBERGEN	10106553	055-2010-15-0594139	055-11-05-000298-8	1,000	1,000	2
406-3945 3536	15/01/2011	BELLAFLOR ROSES AND MORE	10106556	055-2010-15-0594468	055-11-05-000298-8	2,000	2,000	4
406-3945 3536	15/01/2011	BERKELEY BOSTON	10106562	055-2010-15-0594468	055-11-05-000298-8	1,000	0,500	1
724-2970 6814	15/01/2011	BLUMEN GALLIA S.	10100348	055-2010-15-0594360	055-11-05-000298-8	8,750	8,500	19
136-6132 8944	15/01/2011	COMERCIAL ITH 2	10105904	055-2010-15-0594085	055-11-05-012098-8	1,500	1,500	3
136-6432 8867	15/01/2011	EL WESTOSIA DOB 2	10105902	055-2010-15-0594362	055-11-05-012098-8	1,000	1,000	2

Gráfico 114. d1) EXPORTERS - Consulta por Búsqueda de Awb por fecha de Coordinación.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Para poder visualizar el detalle de la guía, el usuario podrá dar clic sobre la guía, y posteriormente visualizará los datos principales de la guía; con los datos de coordinación de la florícola y si es que lo tiene, los datos de ingreso de bodega de la florícola:

Exportador	HAWB	Num Ref	Cajas RSV	Cajas Coord	Num Ing	Cajas Rec	Piezas Rec	Fecha Hora Entrega	Tmp	Observaciones
EXPOFLOR CIA.LTDA.	10106519	055-2010-15-0594139	1,500	1,250	1130410	1,000	2	14/01/2011 - 20:59:29	4,0	ING. FACTURAS EXTERNAS
					1130410	0,250	1	14/01/2011 - 20:59:29	4,0	ING. FACTURAS EXTERNAS

Gráfico 115. e1) EXPORTERS –Visualización del Detalle de Guía.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

La segunda opción *Consulta por Awb por Rango de Fechas de Coordinación*, permitirá realizar la búsqueda de las guías de la florícola en un rango de fechas:

TRANS International CARGO

EXPOFLOR CIA.LTDA. LOGOUT

Wednesday, January 19, 2011

Menu Principal AWB por Fecha de Coordinación AWB por rango de Fechas de Coordinación AWB específica

Servicios en Línea Florícolas - Búsqueda por Rango de Fechas

Fecha de Vuelo Inicial: 15/01/2011 Fecha de Vuelo Final: 19/01/2011

Consultar Embarques

AWB	FECHA VUELO	CLIENTE	HAWB	NUM REFRENDO	NUM MANIFIESTO	BOX RSV	BOX REG	PCS RE
406-3945 3536	15/01/2011	BELLAFLOR BELLE FLORAL	10106519	055-2010-15-0594139	055-11-05-000298-8	1.500	1.250	3
406-3945 3536	15/01/2011	BELLAFLOR C AND C CUT FLD	10106525	055-2010-15-0594458	055-11-05-000298-8	1.500	1.250	3
406-3945 3536	15/01/2011	BELLAFLOR EAST COAST WHOL	10106533	055-2010-15-0594458	055-11-05-000298-8	1.000	1.000	2
406-3945 3536	15/01/2011	BELLAFLOR EMPIRE CUT FLOW	10106755	055-2010-15-0594458	055-11-05-000298-8	0.500	0.500	1
406-3945 3536	15/01/2011	BELLAFLOR FLORAL TRENDS	10106536	055-2010-15-0594458	055-11-05-000298-8	0.500	0.500	1
406-3945 3536	15/01/2011	BELLAFLOR MAHONEY'S GARDEN CENTERS	10106547	055-2010-15-0594458	055-11-05-000298-8	0.500	0.250	1
406-3945 3536	15/01/2011	BELLAFLOR P RAVENSBERGEN	10106553	055-2010-15-0594139	055-11-05-000298-8	1.000	1.000	2
406-3945 3536	15/01/2011	BELLAFLOR ROSES AND MORE	10106556	055-2010-15-0594458	055-11-05-000298-8	2.000	2.000	4
406-3945 3536	15/01/2011	BERKELEY BOSTON	10106562	055-2010-15-0594458	055-11-05-000298-8	1.000	0.500	1
724-2970 6014	15/01/2011	BLUMEN GALLI AG.	10100348	055-2010-15-0594369	055-11-05-000298-8	8.750	8.500	19
136-6133 8944	15/01/2011	COMERCIAL ITH 2	10105904	055-2010-15-0594085	055-11-05-012698-8	1.500	1.500	3
436-6432 8855	15/01/2011	EL HISTORIADOR 2	10105902	055-2010-15-0594085	055-11-05-012698-8	1.000	1.000	2

Gráfico 116. f1) EXPORTERS – Consulta por Awb por Rango de Fechas de Coordinación.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

Posteriormente, el proceso para poder ver las guías será el mismo que se describió anteriormente.

La tercera opción *Consulta por Búsqueda por número de Awb específico*, permitirá realizar la búsqueda de una guía por un número específico:

TRANS International CARGO

EXPOFLOR CIA.LTDA. [LOGOUT](#)

Wednesday, January 19, 2011

Menu Principal | [AWB por Fecha de Coordinación](#) | [AWB por rango de Fechas de Coordinación](#) | [AWB específica](#)

Servicios en Línea Florícolas - Búsqueda por Awb Especifica

Guia Aerea (###-####-####): 406-3945 3536
[Consultar Embarques](#)

Datos de la Guia Aérea

AWB : 406-3945 3536
Aerolinea: UNITED PARCEL SERVICE CO
Cliente: DESTINY FLOWERS
Manifiesto CAE: 055-11-05-000298-8

Información Vuelo

Vuelo	Origen	Salida	Destino	Arribo
383	UIO	15/01/2011 - 12:01	MIA	15/01/2011 - 19:01

Información de Exportador

Exportador	HAWB	Num Ref	Cajas RSV	Cajas Coord	Num Ing	Cajas Rec	Piezas Rec	Fecha Hora Entrega	Tmp	Observaciones
EXPOFLOR CIA.LTDA.	10106533	055-2010-15-0594458	1.00	1.00						
					1130410	0.50	1	14/01/2011 - 20:58:33	4.0	ING. FACTURAS EXTERNAS
					1130410	0.50	1	14/01/2011 - 20:58:33	4.0	ING. FACTURAS EXTERNAS
EXPOFLOR CIA.LTDA.	10106755	055-2010-15-0594458	0.50	0.50						

Ver [100%]

Gráfico 117. g1) EXPORTERS – Consulta por Búsqueda por número de Awb específico.

Fuente: TRANS Internacional Cargo.

Autor: Sánchez Gallardo José.

CAPÍTULO VI.

PLAN DE CONTINGENCIA.

6.1 Definición del plan de contingencia local

La base de datos web (Canadá) es alimentada diariamente cada 30 minutos de la información de la base de datos local (Quito) en la cual los usuarios trabajan diariamente.

La información en la Web debe estar disponible las 24 horas del día, los 365 días del año, solo en casos especiales, al momento de realizar el mantenimiento a los servidores, el último domingo de cada mes, la información no está disponible por 30 minutos.

Para la base de datos local se ha definido el siguiente plan de contingencia:

1. Instalación de un disco duro externo para sacar respaldos de la base de datos local en caso de surgir cualquier imprevisto mientras se está realizado el mantenimiento de los servidores;
2. La base de datos local se respalda cada 1 hora, las 24 horas del día, los 365 días del año;
3. Si no existen novedades, como: pérdida de información o revisión de históricos, de los respaldos de las bases de datos, se borran siempre de los 6 últimos meses;
4. Con respecto al código del sistema, este se lo respalda cada vez que existe algún cambio en el sistema, por petición de los usuarios,
5. Los **log** que se almacenan de la base de datos, son borrados cada mes para que no se acumulen y ocupen espacio en disco;
6. A más del disco duro externo, el servidor donde se encuentra la base de datos local cuenta con discos espejo, en caso de que alguno falle;

7. Si existiera algún daño en el servidor, o éste perdiera su capacidad de funcionamiento por algún daño externo, se tiene un servidor de backup, que tiene la misma estructura del sistema operativo y base de datos, para pasar toda la información a éste servidor de respaldo; y,
8. En caso de que los servicios fallen, se respalda una vez cada 6 meses los archivos de configuración, tanto de la base de datos como de las configuraciones de java.

6.2 Definición del plan de contingencia servidor de aplicaciones remoto

Ya que se tiene acceso físico a la base de datos del servidor de aplicaciones (Canadá), cada último domingo de cada mes se reinicia el servidor siguiendo el siguiente procedimiento:

1. Respaldo de la base de datos web;
2. Limpieza de log almacenados en la ejecución de servicios Apache Tomcat;
3. Reinicio del servidor;
4. Verificación de servicios levantados: postgresql;
5. Verificación de página web funcional; y,
6. Pruebas de funcionalidad con el cliente más grande de la agencia: Destiny Flowers.

En caso de que el servidor remoto fallare, el procedimiento es el siguiente:

1. Se envía un email de solicitud de revisión del servidor remoto (el tiempo de respuesta, máximo es de treinta -30- minutos); y,
2. Si el servidor tiene problemas físicos, se realiza un cambio de servidor, con la misma estructura ya implantada, el personal ESECUREDATA realiza el cambio (el tiempo máximo de respuesta es de dos -2- horas).

En caso de que el **isp** que maneja el proveedor del servidor falle, no se tiene plan de contingencia, por lo que se ha recomendado:

1. Tener alojada la misma estructura del servidor remoto, en un servidor local con un isp local; y,
2. Si falla el servidor remoto, redireccionar las ips remotas a las ips locales.

Es importante mencionar que todo está en función de las políticas locales de la empresa donde se ha desarrollado el sistema.

CAPÍTULO VII.

CONCLUSIONES Y RECOMENDACIONES.

7.1 Conclusiones

- Gracias al desarrollo del “Análisis, Diseño, Implementación e Implantación de un sistema de integración de datos para el rastreo de carga de flores para el sistema TRACKING-CARGO de la empresa ENTICO Cía. Ltda.”, así titulada la presente tesis, las actividades comerciales que realiza la empresa y los usuarios/clientes de la misma se vuelven más fáciles de entender y usar, gracias a la eficiencia, eficacia y practicidad que el sistema ofrece;
- El sistema de rastreo de carga permite a los usuarios: Cliente, Bróker y Florícola verificar la documentación de sus embarques, vuelos que previamente han despegado, usando como opción principal la búsqueda por aquella fecha registrada en la reservación de sus guías; si aún no han despegado sus vuelos, los usuarios pueden comprobar los estados de sus guías aéreas, verificando la cantidad de cajas que se están reservando;
- La agencia de carga que hace uso del sistema, para informar a los usuarios: Clientes, Bróker y Florícolas sobre la documentación, procedían a digitalizar la misma, y a enviarla por cada cliente, bróker y florícola vía email; el sistema en la actualidad permite que los usuarios digitalicen toda ésta información, mediante un servidor a través del servicio web, se permite a los usuarios que descarguen la misma cuando lo necesiten y aquello que les es verdaderamente relevante dependiendo de la categoría que ostenten;
- El cambio de proceso del envío de documentación vía email, a permitir que los usuarios mediante el aplicativo web descarguen la

información de un servidor, logró reducir el tiempo de respuesta en el envío de los documentos por parte de la agencia, puesto que cada usuario puede obtener la información a la hora que necesite, sin pedir a los empleados de la misma enviar los documentos;

- Antes de la implementación del sistema, la agencia de carga previo al vuelo generaba cuadros en Excel enviando unas pre-alertas que contenía la información estimada con las cajas de vuelo, éste proceso era generado por los empleados: actualmente la información se transmite cada 30 minutos a la web, donde los usuarios puede ir verificando sus cajas estimadas, similares a los pre-alertas, sin necesidad de que los empleados de la agencia inviertan tiempo en este proceso y puedan dedicarse a realizar otras tareas;
- La información de la agencia se encuentra en un servidor ubicado en el País de Canadá, por motivos relacionados con los estatutos internos de la empresa y mejor calidad de servicio, el acceso a la información es mucho más rápida, pues desde la implementación del sistema hasta la actualidad, dicho servidor no ha registrado ninguna caída, con éste tipo de proveedores se asegura que la información siempre esté presente para beneficio de los usuarios; sin embargo si existiera alguna caída –en caso hipotético–, dicha información se encuentra en un servidor local, misma que podemos poner a disposición de los usuarios en caso de que el servidor principal fallara;
- El sistema de rastreo tiene procesos que permiten la migración de la información cada 30 minutos, así, sin importar la hora y en cualquier parte del mundo, los usuarios: Clientes, Bróker, Florícolas, pueden siempre acceder a la información actualizada;
- La utilización de herramientas Open Source ha sido primordial, pues se han reducido costos para la agencia de carga, así como también ya gran variedad de complementos que encajan con las mismas

herramientas que permiten al sistema ser escalable y acoplarse con actualizaciones importantes a los sistemas operativos y browser que cada día desarrollan nuevas versiones;

- Para la detección de errores durante los procesos de transferencia de datos mediante Web Services, el sistema ha sido construido para arrojar logs claros que permiten verificar o encontrar un método con error, así mismo el sistema ha sido construido de manera modular, para que un cambio no afecta al resto de procesos; y,
- El uso de la plataforma J2EE ofrece buenas perspectivas de desarrollo para aquellos que deseen basar la arquitectura en productos de software libre, pues tiene soporte en múltiples sistemas operativos; es avalada por empresas como SUN, IBM, ORACLE, etc), es una plataforma madura por la cantidad de años que actualmente tiene, además si se presentase cualquier problema o inconveniente existen foros web en gran cantidad de programadores que pueden ayudar para solventar los problemas que pudiesen surgir con la utilización de éstos sistemas operativos.

7.2 Recomendaciones

- Antes de proceder con cualquier desarrollo de un sistema web, es necesario que se levante toda la información acerca de lo que el cliente necesita, para así tener claro cuál es el alcance que va a tener el proyecto, lo que ayudará a mantener las metas claras y benéficas para el usuario/cliente;
- Es necesario considerar que no siempre las exigencias del cliente van a ser estáticas de ésta manera el sistema desarrollado tendrá tendencia a crecer, por lo que la estructura del sistema debe ser escalable y

desarrollarse de manera progresiva de acuerdo a las actividades y necesidades de los usuarios/clientes;

- Puesto que la base de datos remota es simplemente un repositorio de consultas, es necesario que las tablas construidas sean lo más sencillas posibles, que eviten que el resultado de una consulta demore el procesamiento, además deben ser prácticas y entendibles para obtener la menor cantidad de errores posibles en el ingreso de datos por parte de los usuarios/clientes;
- La estructura del código debe estar orientado a objetos siguiendo los lineamientos considerados en su desarrollo, para que cualquier cambio o inserción de un nuevo código se haga de manera rápida y sin complicaciones;
- El uso de foros de internet es realmente aconsejable ya que estos nos ayudan a buscar soluciones a desarrollos que parecen ser muy complicados;
- Es necesario no depender de un proveedor externo para el manejo de la base de datos web, ya que cuando se dan inconvenientes, no se puede manipular u observar dichos problemas de manera local; y,
- Es necesario centralizar el código en un repositorio de código para llevar un versionamiento de cambios y en caso de cambios otros programadores sepan que cuales fueron los últimos cambios.

7.3 Proyección a futuro

- El sistema ha sido construido de tal manera que el usuario pueda valorar la posibilidad de extender el mismo agregando, cambiando o modificando módulos que no solo permitan la consulta de los embarques sino que también permitan la inserción de datos, o la

gestión de determinados procesos que son propios de Fincas o Clientes; por ejemplo:

1. Los números de la Declaración Aduanera de Exportación (DAE), son utilizadas por las fincas con la finalidad de justificar sus exportaciones en aduanas, mismo que puede registrarse con total seguridad en el sistema web, para que la agencia, dentro de su proceso de coordinación y generación de embarques diarios, ya no tenga la necesidad de solicitar a la finca dicho número, sino que, mediante consultas el usuario solo se encargue de revisar y verificar el ingreso del mismo;
2. El ingreso de la coordinación diaria; en este proceso la finca puede realizar sus reservas o asignaciones de cajas a clientes que previamente ya hayan sido reservados en la agencia de carga; en este proceso la finca ya no tendría que enviar o comunicarse con la agencia para coordinar las cajas a ser transportadas, sino que directamente en el sistema web lo podría realizar; tomado en cuenta que dichas coordinaciones generadas en la web deberían pasar por un control dentro de la agencia donde se autorice después de una revisión, las coordinaciones que hayan sido generadas en el sistema;
3. La generación de las guías de finca o HAWB, actualmente la agencia se encarga de generar estos documentos y enviarlos a las florícolas, sin embargo, como ya se tiene toda la información necesaria, se puede agregar un módulo para que las fincas puedan directamente descargar este documento para su uso, y en caso de necesitar certificación, únicamente podrían solicitar en la agencia de carga la firma y sello para certificar este documento;
4. Actualmente para que los usuarios ingresen al sistema, cada uno consta de un usuario y una clave para hacerlo, pero, ya que actualmente se está impulsando en Ecuador el uso del dispositivo digital token, que tienen almacenadas claves criptográficas como firmas digitales (Impulsado por

Aduanas), se podría investigar la posibilidad de permitir el ingreso al sistema mediante el uso de este dispositivo, con el fin de agregarle mayor seguridad al momento de ingresar al sistema;

5. Visualización por parte de los gerentes administrativos de la agencia de carga de los datos como: tarifas, pesos, costos cobrados de las guías que ya han volado, etc; con lo cual, ya no es necesario estar en el sistema local para verificar éstos datos, sino directamente lo podrían hacer desde la web, sin olvidar que en estos casos se debe administrar otro perfil de usuario para las consultas de dichos datos;
6. Generación y visualización de información tanto para Clientes, Brokers, y Florícolas de datos que les permitan observar guías con pesos, piezas, tarifas, y totales para consultas de los usuarios, con opción de exportación a Microsoft Excel, este proceso puede ayudar a los usuarios a llevar datos estadísticos de sus empresas; y,
7. Se tiene previsto investigar nuevas tecnologías que permitan a este sistema ser utilizado en dispositivos móviles como celulares, ipad, tablets, etc; con el fin de poder tener el sistema funcionando bajo cualquier plataforma de hardware y software:
 - Es posible poder recrear todo el proceso que realizan las agencias de carga conjuntamente con la florícola, en la web, ya que esto ayudaría a no depender de sistemas centralizados en las agencias de carga sino que desde cualquier lugar se lo podría hacer, sin embargo es necesario revisar los procesos se pueden implementar y cuales no, para poder facilitar el manejo de la exportación tanto a fincas, clientes, bróker y especialmente la agencia de carga.

GLOSARIO DE TÉRMINOS

1. Fitos Sanitarios: Es el documento probatorio de la condición de salubridad en que se encuentran determinados productos vegetales exportables. En Ecuador los certificados fitosanitarios son otorgados por las dependencias del Ministerio de Agricultura y Ganadería y no son obligatorios a efectos de autorizar las exportaciones sino que deben ser obtenidos por los exportadores a efectos de cumplir regulaciones sanitarias y de salubridad vigentes en los mercados de exportación.

2. Brókers: Son creadores de mercados: ellos atraen a compradores y vendedores y facilitan las transacciones. Los Brókers frecuentemente juegan un rol en los mercados de negocio a negocio, negocio a consumidor, consumidor a consumidor. Usualmente un Bróker cobra una cuota o comisión por cada transacción que lleva a cabo al consumidor, consumidor a consumidor. Usualmente un Bróker cobra una cuota o comisión por cada transacción que lleva a cabo.

4 Paletizadora: Máquina paletizadora o Estibadora. Se trata de una máquina que combina componentes mecánicos y eléctricos con la finalidad de colocar productos generalmente almacenados en cajas, sacos, tambores, entre otros, sobre un palé, que puede ser de madera, metal o plástico para la conformación de una estiba.

En sustitución del palé, existen también las hojas deslizables (también conocidas como "slip sheets"), que pueden ser de cartón o plástico. Para poder colocar las estibas una encima de la otra usando estas hojas, hay que instalar en los montacargas un aparato especial que tiene por nombre "push and pull".

Las paletizadoras de sacos de 25 kg actuales pueden paletizar desde 600 sacos por hora (con robot o brazo mecánico) hasta 4.500 sacos por hora (generalmente máquinas un poco más complejas utilizadas en la industria del cemento).

5. Iterativo: Término que indica una acción repetitiva; En programación, Iteración es la repetición de un proceso dentro de un programa de computadora. Puede usarse tanto como un término genérico (como sinónimo de repetición) así como para describir una forma específica de repetición con un estado mutable.

Cuando se usa en el primer sentido, la recursividad es un ejemplo de iteración, pero que usa su propia notación (notación recursiva), que no es el caso de iteración.

Sin embargo, cuando se usa en el segundo sentido (caso más restringido), la iteración describe el estilo de programación usado en lenguajes de programación imperativa. Esto está en contraposición de la recursividad, la cual tiene un enfoque más declarativo.

6 HAWB: House Air Waybill, Conocimiento de embarque aéreo emitido por el expedidor de un grupaje. Cada exportador recibe del agrupador un documento detallando su envío en el grupaje global. Este documento emitido por el agrupador y en su nombre, es el HAWB (House Air Waybill). En caso de créditos documentarios, el HAWB es asimilado al Air Waybill AWB original, en la medida en que el emisor asume una responsabilidad en calidad de transportista o en calidad de agente de un transportista designado.

7 AWB: Es el contrato de transporte aéreo para mercancías; Definición: Este documento es emitido por la compañía de transporte aéreo, donde prueba la existencia

de un contrato de transporte, acredita que el transportador ha tomado la carga obligándose a entregarla en el aeropuerto convenido previamente.

8 Open Source: Open source es una expresión de la lengua inglesa que pertenece al ámbito de la informática. Aunque puede traducirse como “fuente abierta”, suele emplearse en nuestro idioma directamente en su versión original, sin su traducción correspondiente.

Se califica como open source, por lo tanto, a los programas informáticos que permiten el acceso a su código de programación, lo que facilita modificaciones por parte de otros programadores ajenos a los creadores originales del software en cuestión.

12 Quarts: Es uno de los más completos y destacados proyectos en la plataforma Java, cuenta con un API robusta pero de fácil integración y es OpenSource (bajo licencia Apache 2.0). Puede ser utilizado tanto en aplicaciones Java SE con POJOs así como en aplicaciones Java EE, en dónde se puede integrar dentro del ciclo de vida de una transacción JTA, además se puede configurar para trabajar en clusters.

Además de las características mencionadas cabe destacar la integración que ofrece Spring framework para la simplificación de la programación de tareas Quartz. Los principales conceptos utilizados en Quartz son Job, Trigger y Scheduler, los cuales ejemplificaremos a continuación.

13 XML: Siglas en inglés de eXtensible Markup Language ('lenguaje de marcas extensible'), es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C). Deriva del lenguaje SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje

definido por SGML) para estructurar documentos grandes. A diferencia de otros lenguajes, XML da soporte a bases de datos, siendo útil cuando varias aplicaciones se deben comunicar entre sí o integrar información. (Bases de datos Silberschatz).

XML no ha nacido sólo para su aplicación para Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable.

XML es una tecnología sencilla que tiene a su alrededor otras que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores. Tiene un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.

14 Asequibles: Que se puede alcanzar o conseguir sin mucha dificultad.

15 Parametrización: El concepto parametrización, responde a una cuidadosa planificación de la biblioteca a la hora de determinar, que tipologías de documentos introduce, que políticas de préstamo afectarán a qué tipo de documentos y qué tipos de usuarios, qué denominación tendrán las localizaciones de los documentos, etc. Todos estos valores, responden ante lo que se suele llamar parametrización del sistema.

Si bien no es algo que venga hecho, sí es responsabilidad del centro elaborar dicha planificación y establecer unos parámetros normalizados para operar con la información y registros con los que se alimente el sistema. En este apartado se pretende aportar un método de trabajo para identificar rápidamente qué aspectos son

susceptibles de un proceso de parametrización y en que aspectos pueden ser definidos.

16 Paradigma: El término paradigma significa «ejemplo» o «modelo». En todo el ámbito científico, religioso u otro contexto epistemológico, el término paradigma puede indicar el concepto de esquema formal de organización, y ser utilizado como sinónimo de marco teórico o conjunto de teorías.

En los tiempos modernos, este concepto fue originalmente específico de la gramática; en 1992 el diccionario Merriam-Webster definía su uso solamente en tal contexto, o en retórica para referirse a una parábola o a una fábula. En lingüística, Ferdinand de Saussure ha usado paradigma para referirse a una clase de elementos con similitudes. El término tiene también un significado en el terreno de la psicología refiriéndose a aceptaciones de ideas, pensamientos, creencias incorporadas generalmente durante nuestra primera etapa de vida que se aceptan como verdaderas o falsas sin ponerlas a prueba de un nuevo análisis.

17 Mapeo: Término que designa colectivamente los distintos procedimientos (tanto genéticos como físicos) empleados en la construcción de los mapas génicos.

BIBLIOGRAFÍA

1. Cáceres Tello, J. (2 de Junio de 2012). Dpto. Ciencias de la Computación. Recuperado el 20 de Enero de 2013, de Diagramas de Secuencia: <http://www2.uah.es/jcaceres/capsulas/DiagramaSecuencia.pdf>.
2. Cruz, A. (01 de 07 de 2012). OFIMATICA II - Turno Vespertino. Recuperado el 12 de 11 de 2012, de OFIMATICA II - Turno Vespertino: <http://sandycmts.blogspot.com/2013/01/diagrama-de-flujo.html>
3. Latina, O. (21 de 12 de 2011). Menú Curso Java Web (JSP's/Servlets). Recuperado el 12 de 11 de 2012, de Estructura (Framework) "MVC" ("Model - View - Controller"): <http://javaweb.osmosislatina.com/curso/mvc.htm#filtros>
4. Library, M. (1 de Enero de 2013). MSDN Library. Recuperado el 2 de Enero de 2013, de MSDN Library: <http://msdn.microsoft.com/es-es/library/x9fsa0sw.aspx>
5. Magaña, E. (01 de 01 de 2011). Administrando Proyectos de Software I.S.C. Recuperado el 01 de 01 de 2013, de Un panorama general de la Ingeniería de Proyectos de Software: <http://administrandoproyectos.blogspot.com/2011/01/rational-unified-process-rup.html>
6. MasterMagazine. (5 de Enero de 2012). MasterMagazine. Recuperado el 7 de Febrero de 2013, de MasterMagazine: <http://www.mastermagazine.info/termino/5400.php>
7. Olade, K. (01 de 01 de 2011). Cap. 1: Introducción a la dirección de Proyectos. Recuperado el 23 de 02 de 2012, de PMBOK (resumen norma nacional)

americana ANSI/PMI 99-001-2004):

<http://www.ehu.es/Degypi/PMBOK/cap1.htm>

8. Rodríguez, J. (01 de 01 de 2012). Un Aporte A Java Un Pequeño Aporte. Recuperado el 04 de 10 de 2012, de Para Los Que Inician En Java:
http://ungranoparajava.blogspot.com/2009_05_01_archive.html
9. Sánchez, M. (01 de 01 de 2012). JAAS (Java Authentication and Authorization Service) y Jboss. Recuperado el 14 de 11 de 2012, de JAAS (Java Authentication and Authorization Service) y Jboss:
<http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=jaas>
10. Tecnoquo. (08 de 03 de 2009). Qué es el CSS y para que sirve. Recuperado el 20 de 10 de 2012, de CSS: <http://www.tecnoquo.com/ecss-y-los-estandares-web/>
11. Vargas M, a. d. (01 de 06 de 2011). Blogspot. Recuperado el 20 de 11 de 2012, de ANÁLISIS DE SISTEMAS:
<http://antimaiyenud.blogspot.com/2011/06/uml.html>.
12. Vélez Amelia. (01 de 06 de 2012). Blog Para Sistemas de Información. Recuperado el 12 de 11 de 2012, de Blog Para Sistemas de Información:
<http://amaly88.blogspot.com/>
13. Villanueva, A. (16 de 07 de 2008). MADEJA. Recuperado el 04 de 11 de 2012, de RichFaces: <http://es.scribd.com/doc/51494264/RichFaces>
14. Wikipedia. (s.f.). Wikipedia.org. Obtenido de Wikipedia.org:
http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado.